ROCKWELL SOFTWARE

Arena 3DPLAYER®

User's Guide

Publication AREN3D-UM001C-EN-P–July 2005
Supersedes Publication AREN3D-UM001B-EN-P

Rockwell Automation

Contacting Rockwell Software

Technical Support Telephone—1.440.646.5800
Technical Support Fax—1.440.646.5801
World Wide Web—www.software.rockwell.com

Copyright Notice

© 2005 Rockwell Software Inc. All rights reserved. Printed in the United States of America.

Portions © 2005 Systemflow Simulations, Inc.

This manual and any accompanying Rockwell Software products are copyrighted by Rockwell Software Inc. Any reproduction and/or distribution without prior written consent from Rockwell Software Inc. is strictly prohibited. Please refer to the license agreement for details.

Commercial runtime models may be legally loaded and run only by employees of organizations with a commercial Arena license. Models created using a research licenses may not be used for commercial use. Any other use of a runtime model is illegal and unauthorized.

Commercial Arena software can be obtained by contacting Rockwell Software at 1.412.741.3727 or contacting your local representative (listed under Partners at www.ArenaSimulation.com <http://www.ArenaSimulation.com>).

Trademark Notice

Arena and Arena 3DPlayer are registered trademarks and the phrase “Forward Visibility for Your Business” and the Rockwell Software logo are trademarks of Rockwell Automation.

AutoCAD is a registered trademark of Autodesk, Inc. 3ds Max is a registered trademark of Discrete, a wholly owned subsidiary of Autodesk, Inc. Maya is a registered trademark of Alias Systems, a division of Silicon Graphics Limited. Pro/ENGINEER is a registered trademark of Parametric Technology Corporation. Windows is a registered trademark of Microsoft Corporation. Vizx3D copyright by Virtock Technologies, Inc. Cinepak is a registered trademark of Radius, Inc. DivX is a registered trademark of DivXNetworks, Inc.

All other trademarks and registered trademarks are the property of their respective holders and are hereby acknowledged.

Warranty

This Rockwell Software product is warranted in accord with the product license. The product’s performance will be affected by system configuration, the application being performed, operator control, and other related factors.

This product’s implementation may vary among users.

This manual is as up-to-date as possible at the time of printing; however, the accompanying software may have changed since that time. Rockwell Software reserves the right to change any information contained in this manual or the software at anytime without prior notice.

The instructions in this manual do not claim to cover all the details or variations in the equipment, procedure, or process described, nor to provide directions for meeting every possible contingency during installation, operation, or maintenance.

Contents

1 • Welcome to Arena 3DPlayer 1

What is Arena 3DPlayer? 1
Intended audience 1
Where can I go for help? 1
Reference the user's guides 1
Explore the examples 1
Get help online 2
Get phone support 2
Get Web support 2
Get training 2
Get consulting services 3
Contact us 3

2 • Getting Started 5

The Arena 3DPlayer environment 5
Menus and commands 6
Layout and playback files 9
Tutorial 1: Opening an existing animation 10
Navigating in the 3D pane 11
Named Views 12
Tutorial 2: Navigating in 3D 13
Run mode 13
Tutorial 3: Running the animation 14

3 • 3D Animation 15

Animation methods 15
Working with the Layout Editor 16
The Tree View pane 17
The 3D pane 19
The Edit pane 19
Adding 3D objects 20
Resources 20
Queues 22
Globals 23
Storages 24
3D Levels 25

Entity pictures 30
Tutorial 4: Building a simple animation 31
Manipulating 3D shapes 33
The Edit pane 34
Using the 3D grid 37
3D multiple selecting and editing 39

4 • Shapes and Static Systems 41

Working with the Shape Manager 41
Default shapes 43
Selecting the target shape 43
Selecting the new shape 44
Making the assignment 44
Duplicate shapes 45
Shape libraries 45
Using shapes from outside the standard library 46
Tutorial 5: Enhancing the simple animation 47
Shape editing 47
Shape Editor basics 48
Adding and organizing components and groups 51
About VRML files 51
Which VRML files will import? 52
Other drawing tools for shapes 53
Static systems 53
Creating and populating a static system 54
Importing a CAD layout 54
Adding shapes to a static system 56
Static shapes from Arena 56
3D wall systems 56

5 • The Dashboard 59

Navigating in the Dashboard pane 59
Adding Dashboard objects 61
Clocks 61
Variables 62
Images 64
Levels 65
Text 66
Plots 67
Histogram 69

More about the Dashboard 70
Editing objects 70
Other Dashboard facts 70
Tutorial 6: Editing the Dashboard display 71

6 • Material Handling 73

Routes 73
Segments 74
Transporters 76
Adding a transporter 76
Ride points 78
Distances (for free-path transporters) 78
Networks (for guided transporters) 80
Directionality 81
Stations 82
Intersections 83
Parking areas 84

7 • Path Drawing and Editing 85

Drawing a path 85
Residence-type paths 85
Guide-type paths 87
Editing a path 88
Editing path points 88

8 • Importing Arena Animation Information 91

General import considerations 91
Dashboard import considerations 93
Tutorial 7: Importing and running an Arena animation 94
Working with resources 94
Working with seize areas 95
Working with queues 95
Working with entity shapes 95
Analyzing the animation 96
What else can I do with this animation? 96

9 • Enhancing an Animation 97

Live components 97
Static components 98

10 • Creating AVI Movie Files 99

About AVI capture 99
Capturing an animation. 99
Varying the viewpoint 100
Recording quality 101

A • Additional Tips 103

Locating an object. 103
Forward motion 103
Naming conventions 103
Unusual rotation axes 104
Status expressions 104
Distance units 105
Lighting. 105

B • Shape Drawing: Beyond the Basics 107

Tutorial 8: Extrusions 107
Tutorial 9: Revolutions 113

C • Keyboard Controls for Camera Movement 119

Index 121

Welcome to Arena 3DPlayer

What is Arena 3DPlayer?

Arena 3DPlayer® is a powerful post-process tool that provides the ability to create and view 3D animations of your Arena models. You use Arena 3DPlayer to create a 3D animation layout and Arena to create a simulation history for a particular simulation run. Arena 3DPlayer takes these two data sets and produces a smooth, accurate 3D animation of that simulation run.

Intended audience

We assume that you have a familiarity with the concepts and creation of simulation models in Arena before you begin to create animations with Arena 3DPlayer.

Where can I go for help?

Our commitment to your success starts with the suite of learning aids and assistance we provide for Arena 3DPlayer. Even if you have not created 3D animation before, you'll quickly feel at home with Arena 3DPlayer.

Reference the user's guides

The Arena 3DPlayer product package includes this copy of the *Arena 3DPlayer User's Guide*. Information on modeling with Arena can be found in the user's manuals that accompany the Arena products, both in print and electronic form on the product CD.

Document conventions

Throughout the guide, a number of style conventions are used to help identify material. New terms and concepts may be emphasized by use of italics or bold; file menu paths are in bold with a (>) separating the entries (e.g., go to **File > New Layout**); text you are asked to type is shown in Courier Bold (e.g., in this field, type **`Machine 1`**), and dialog button names are shown in bold (e.g., click **OK**).

Explore the examples

Arena 3DPlayer is accompanied by example files that illustrate 3D animation. For a description of and list of Arena 3DPlayer's examples, go to **Help > Arena 3DPlayer Help**.

Get help online

Online help is always at your fingertips! For a view of the help topics, go to **Help > Arena 3DPlayer Help** and explore the table of contents.

Get phone support

Rockwell Software provides full support for the entire Arena family of products. Questions concerning installation, how modules work, the use of the model editor, and the use of the software are handled by technical support.

ARENA TECHNICAL SUPPORT INCLUDES:

- (for users on active maintenance) a technical support hotline and e-mail address staffed by full-time, experienced professionals
- help with installation problems or questions related to the software's requirements

If you call the support line (1.440.646.5800), you should be at your computer and be prepared to give the following information:

- the product serial number
- the product version number
- the operating system you are using
- the exact wording of any messages that appeared on your screen
- a description of what happened and what you were doing when the problem occurred
- a description of how you tried to solve the problem

Get Web support

In addition to phone support, the Rockwell Automation Customer Support Center offers extensive online knowledgebases of tech notes and frequently asked questions for support of non-urgent issues. These databases are updated daily by our support specialists.

To receive regular e-mail messages with links to the latest tech notes, software updates, and firmware updates for the products that are of interest to you or to submit an online support request, go to http://support.rockwellautomation.com.

And be sure to check the Arena User Zone section of our Web site at www.ArenaSimulation.com. The User Zone links to a peer-to-peer forum on Arena topics and has a link to a download page where you can check for possible software updates (patches). If you can't find the answer you need, contact your local representative or Arena technical support.

Get training

Do you need training? Rockwell Software offers a standard training course comprised of lecture and hands-on workshops designed to introduce you to the fundamental concepts of modeling with Arena.

We also offer customized training courses designed to meet your specific needs. These courses can be held in our offices or yours, and we can accommodate one person or twenty. You design the course that's right for you! Simply contact our consulting services group to discuss how we can help you achieve success in your simulation efforts.

Get consulting services

Rockwell Automation provides expert consulting and turnkey implementation of Arena, covering the entire product suite. Please call your local representative for more information or send email to Arena-Info@ra.rockwell.com.

Contact us

We strive to help all of our customers become successful in their process-improvement efforts. Toward this objective, we invite you to contact your local representative or Rockwell Software at any time that we may be of service to you.

Support E-mail: Arena-Support@ra.rockwell.com
Corporate E-mail: Arena-Info@ra.rockwell.com
Support Phone: 1.440.646.5800
URL: www.ArenaSimulation.com
URL: www.software.rockwell.com

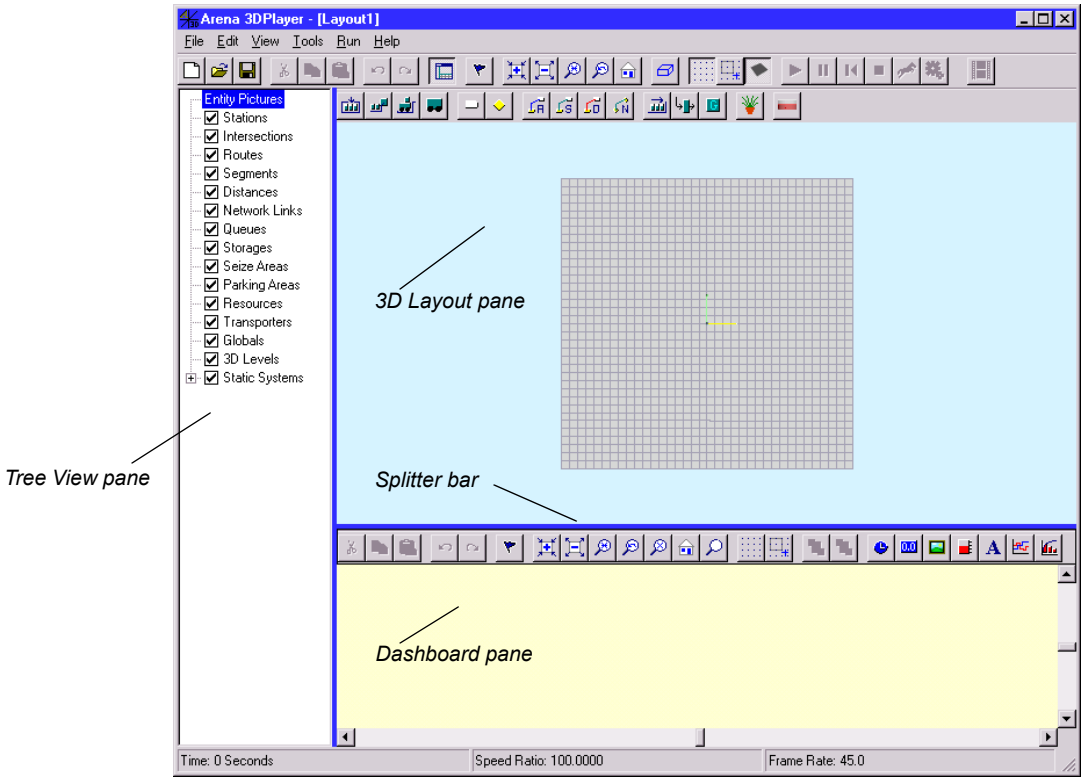
2 Getting Started

Welcome to Arena 3DPlayer—a flexible, Windows®-oriented program designed to help you create 3D animations of your Arena models.

If you are a new user of Arena, you should spend some time becoming familiar with Arena before attempting to learn Arena 3DPlayer. This documentation assumes that you are familiar enough with Arena to run Arena and build simple Arena models.

The Arena 3DPlayer environment

The Arena 3DPlayer environment is designed to be familiar to Arena users. The program opens from the Windows Start menu by selecting **Programs > Rockwell Software > Arena 3DPlayer**. The opening screen is shown below.



Below the Windows-style menu and toolbars are the three main regions of the workspace. The left-hand pane contains the Tree View listing of the animation primitives or components. The right-hand pane is the display area of the 3D animation. When a new window first opens, the display is shown in a top-down view. Beneath the 3D display area is the Dashboard. This area is blank when you open a new window, but is the region where two-dimensional information is commonly displayed.

When your animation is complete, you may find that you need to adjust the amount of space occupied by the 3D pane or the Dashboard. Simply click and drag the splitter bar between these panes to modify the panes' sizes. This adjustment will be saved with the layout.

The toolbars for the Dashboard pane and the 3D pane operate separately, even though there are some equivalent commands on both. Button selections must be made based on the pane in which the actions must occur. Examples include the Cut, Copy, Paste toolbar or the View toolbar.

The right-click shortcut menu is pane-specific and contains commands that relate to the actions of the Dashboard or the 3D pane, respectively. The command list switches automatically, depending on which object is selected. Likewise, the Edit menu commands from the menu bar and hotkey assignments switch silently depending on what object you have selected and work as you would expect, regardless of which pane's object is selected.

The next section takes a look at each of these areas so you can learn more about them.

Menus and commands

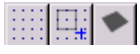
Your tour begins by identifying the various commands on the menus and toolbars in the workspace.

The top row contains six menus with commands arranged similarly to other Windows-based programs. These menus are:

Menu	Description
File	Performs functions similar to other Windows programs with commands for opening, saving, and importing files and for exiting the 3DPlayer program. In addition, the four most recently opened files are listed.
Edit	Contains typical options applied to objects. Here you'll find common commands for Undo, Redo, Cut, Copy, Paste, Delete, Select All.
View	Provides control over the information that is displayed in both the 3D pane and the Dashboard.
Tools	Contains menu equivalents of the toolbar buttons for adding objects to your animation; controls for shape assignment, shape editing, and AVI creation; as well as a dialog for layout settings.

Menu	Description
Run	Controls a running animation.
Help	Offers access to a variety of help options, including online help that is provided by Table of Contents, Index, or Search methods; release notes; 3D navigation tips; manuals online; technical support via Web access; and information about copy protection.

The top toolbar row contains buttons for performing many basic file operations, as well as buttons common to controlling the view of the model display and to control the animation run. Many of these buttons are also represented by commands on the menu bar. These buttons are:

Buttons	Commands
	File New Layout, File Open Layout, File Save Layout
	3D Cut, Copy, Paste
	3D Undo, Redo
	Toggle Split Screen (to hide or show the Dashboard)
	3D Named Views
	3D Zoom In, Zoom Out, View All, View Previous, Home View
	Wire Frame (toggle key to change from filled to wire-frame view)
	3D Grid, 3D Snap to Grid, Show/Hide Floor
	Animation Go, Animation Pause, Animation Start Over, Animation End, Jump To Time, Speed Ratio
	AVI Capture

The 3D pane in Layout view contains its own toolbar, which includes buttons similar in appearance to the equivalent animation buttons in Arena as well as buttons to add static shapes to your display. These are:

Buttons	Commands
	Add Storage, Add Seize Area, Add Parking Area, Add Transporter
	Add Station, Add Intersection
	Add Route, Add Segment, Add Distance, Add Network Link
	Add Queue, Add Resource, Add Global
	Add Static Shape
	Add 3D Level

Note that the toolbar buttons displayed within the 3D pane may be different or may not display at all depending on the activity. For example, the buttons shown in the table above represent those available while in Layout mode; however, a different set of buttons is displayed while in Shape Edit and no buttons display in the 3D pane during a run.

The Dashboard contains its own toolbar to manage the 2D pane and the information contained within it. Many of the buttons operate in the same way as the comparable buttons in the 3D pane, but the control is limited to objects within the Dashboard. Also included here are buttons capable of enhancing your animation with objects like clocks, levels, and static text. The buttons on this toolbar are:

Buttons	Commands
	Dashboard Cut, Copy, Paste
	Dashboard Undo, Redo
	Dashboard Named Views

Buttons	Commands
	Dashboard Zoom In, Zoom Out, View All, View Previous, Show Zoom Percentage, View Home, View Region
	Dashboard Grid, Dashboard Snap to Grid
	Bring to Front, Send to Back
	Add Clock, Add Variable, Add Image, Add Level, Add Text, Add Plot, Add Histogram

Arena 3DPlayer is an animation builder, not a simulation builder. You will find equivalents for essentially everything used in Arena to create “two-step” animations. You won’t find equivalents for any of the Arena tools used to construct simulation models. Model construction is only performed using Arena.

Layout and playback files

Running an animation requires two files—a playback file (*filename*.pbf) and a layout file (*filename*.a3d). The playback file is generated in Arena from software versions 7.01 or higher that are running with valid playback activation installed. The layout file is created in Arena 3DPlayer.

Arena 3DPlayer supports both text and binary playback files generated from within Arena. Binary files are smaller, which can be an advantage when storing or sharing complex animations, and they enable improved animation performance in some circumstances (such as when running complex animations at high speeds).

While it is possible to create a layout first, it is strongly recommended that you first generate the Arena playback file and then create the layout. Following this processing order allows you to take advantage of the Arena names that are included in the playback file (this topic is covered in more detail in the discussion on creation of the layout).

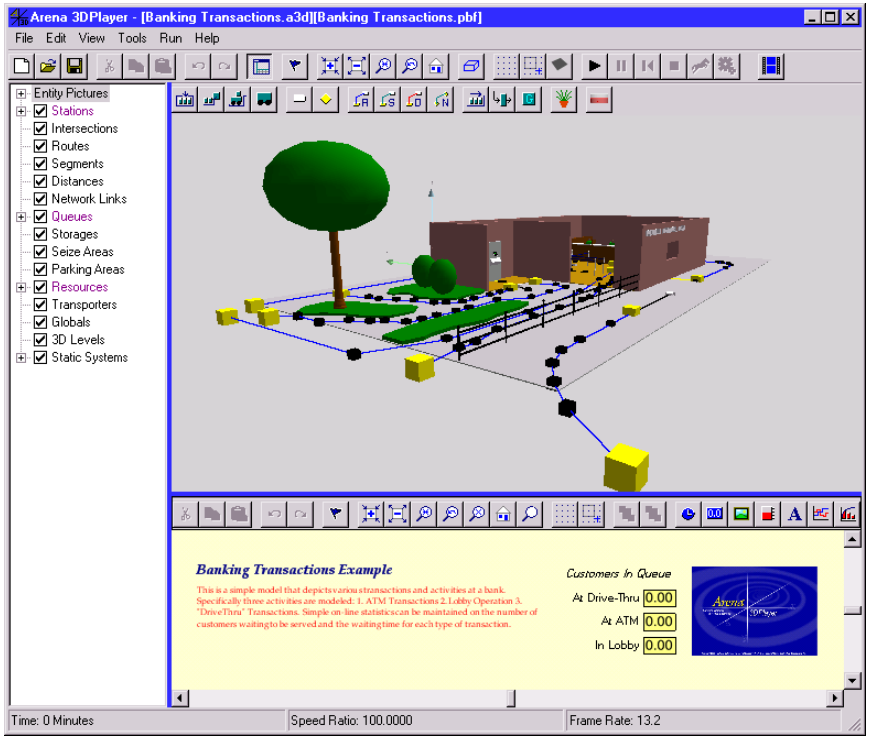
The File menu offers a choice of command options for opening files. However, when the base file names for the layout and playback are the same, you can open them simultaneously by choosing **Open Playback And Layout** and selecting either file. If a file or file pair has been opened recently, the name will appear on the Recent Files list and the file may be re-opened simply by clicking on the file name.

Arena 3DPlayer permits only one layout to be open at any given time. If you wish to close your current layout without exiting the program, simply select **File > New Layout**. You will be prompted first to save your current file, and then the file will close and a new blank layout will appear.

Tutorial 1: Opening an existing animation

Before you create your own animation from an Arena model, you can explore one of the examples from the Arena 3DPlayer program folder to understand how to move around the environment and look at a display of a finished animation.

1. Select **File > Open Playback And Layout** and browse to find the Banking Transactions animation (**`Banking Transactions.a3d`**) in the Examples folder of Arena 3DPlayer.



You'll notice that the window title bar shows that both the layout and the playback files are active concurrently. In the Tree View pane, you see the list of objects in the model. The 3D pane displays the banking animation and the Dashboard pane has a title, some text display, and a graphic image.

2. Keep this file open so you'll be ready for Tutorials 2 and 3 later in this chapter.

Navigating in the 3D pane

You can navigate to view the animation from different angles, at different places, and at different sizes. To change the orientation, size, and center point, you normally use the mouse. Remember that the distance of your mouse movement governs the degree of change. Here are the mouse movements for 3D navigation:

- Drag left to rotate the geometry to the left, and drag right to rotate the geometry to the right.
- Drag up/down to tilt the geometry up/down.
- Right-click and drag up/down to make the geometry bigger or smaller (or use the zoom buttons in the toolbar).
- Click the middle button (or SHIFT+right-click, which comes in handy if you are using a two-button pointing device) to establish a new center point (this also adjusts the center of rotation of the drawing).

  Note: A "wheel mouse" may have a middle button that you click by pressing on the wheel.

Arena 3DPlayer uses a "move the geometry" (not a "move the viewpoint") paradigm for 3D navigation. Of course, internal to the software, the geometry is not actually moving; only the viewpoint is moving. But the mouse action is designed to seem to maneuver the geometry to where you want it, not to maneuver yourself into a different viewing position. (For a list of the keyboard equivalents for viewpoint movement, see Appendix C—"Keyboard Controls for Camera Movement" on page 119.)

In case you get lost in 3D, Arena 3DPlayer provides convenient menu access to the six orthogonal, or right-angled, pre-defined named views: Top, Bottom, Front, Back, Left, Right. These are available by choosing **View > Normal Views**. You may also pan forward/backward and left/right using the directional arrows on your keyboard in a manner that resembles an overhead "flyover" in the chosen direction.

You can also return to the Home view at any time by clicking either the **Home** toolbar button or selecting **View > Home > Show Home View**. You can change the Home View for a given layout by choosing **View > Home > Make This the Home View**. If you then save your file, the new Home View will be stored in the layout file.

Other View control options are available in the upper 3D toolbar. Simply hover your mouse pointer over each of the buttons on the toolbar to display the ToolTip identifying each one, and experiment with their functions. They include:

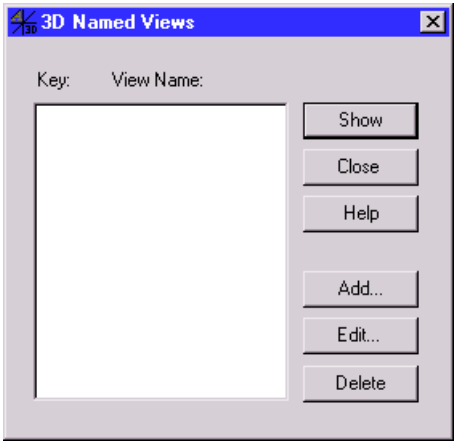
- Toggle Split Screen
- Named Views (built-in and custom)
- Zoom In
- Zoom Out

- View All (Zoom-to-fit)
- View Previous
- Home View
- Wire Frame
- Grid
- Snap to Grid
- Show/Hide Floor

Named Views

The Named Views feature allows you to define custom named views. These may be useful for large, complex animations where it is routine to need to jump to views of specific areas in 3D. They may also be useful in the Dashboard if you have built extensive statistical or other types of displays.

When you choose Named Views, the 3D Named Views dialog appears, as shown below.



Named Views that have been defined appear in the View Name item list. If you click the **Add** button to define a new Named View, you will be prompted to give the view a name. Entering a name and clicking **OK** associates the current view with that name.

If a Named View is selected in the list, you can **Edit** its name or **Delete** it. You can also click **Show**, which switches to that Named View and closes the dialog box.

Dashboard Named Views are defined, used, and saved separately from 3D Named Views, but the dialog box looks identical (except the dialog title is Dashboard Named Views) and the functionality is the same.

Tutorial 2: Navigating in 3D

1. Return to the Banking Transactions layout and experiment with navigation in the 3D pane as described in the previous section.
2. After you have learned to zoom, rotate, and pan in 3D, try defining and using some Named Views to represent different areas of the model.

Run mode

Run mode allows you to view the 3D animation of your Arena model in action; however, this action is only accessible if a playback file is open. With playback and layout files open, select **Run > Go** or simply press the **Go** button (▶) on the toolbar, which causes the animation to run.

You will notice that changing modes causes the status of many toolbar buttons to change from unavailable to available and vice versa.

To Run an animation, you must have a playback file open.

While an animation is running, the clock value is displayed in the left portion of the status bar (at the bottom of the window). The status bar also shows the speed ratio and frame rate of the run. At any time during a run, you may **Pause** or re-start (**Go**) the animation by pressing the buttons from the toolbar. Pressing the **Start Over** button during a pause causes the animation to return to Time 0. The **End** button stops the animation run and returns the screen to the Layout view.

The animation run speed may be changed by pressing the **Speed Ratio** button. The Speed Ratio determines the ratio of animation time to viewing time. If the speed ratio is set to 60, then you are running at exactly 60 times real time. A shortcut method of adjusting the run speed is to use the “>” and “<” keys (the current speed ratio multiplied by 1.25 and .8, respectively).

You can redefine the time frame of the animation by pressing the **Jump To Time** button to jump to a time value that is in the future. The units are the same as the time units defined in your Arena model. If you specify a past time (that is, a value less than the clock value displayed in the lower-left corner), Arena 3DPlayer will will display a warning and ask you to restart the animation and advance to the desired time.

During the animation run, you can attach the observation viewpoint to an entity in the animation by clicking **View > Attach Camera**. Upon clicking **Attach Camera,** the pointer changes to a cross hair, allowing you to click on any entity to position the camera view from that location. You must then press the **Go** button to resume the run. The viewpoint will remain fixed in that position until the entity is destroyed, you choose **View > Detach**

Camera, you press the **ESC** key, or the run ends. During the time that the camera is attached to an entity, normal viewpoint movements by mouse or keyboard are suspended. If you attach the camera to an entity that is later disposed, the viewpoint will automatically return to its position prior to the Attach Camera action.

Tutorial 3: Running the animation

If the Banking Transactions layout and playback files are still open in your workspace, the buttons associated with running the animation can be selected. Choosing **Run > Go** or pressing the **Go** button from the toolbar begins the 3D animation activity.

1. Run the Banking Transactions animation.
2. During the run, pause the animation and make some changes to the speed ratio and the animated time. Then restart the animation to see the effect of the changes. You will notice that when jumping forward in time, the simulation time displayed on the status bar continuously updates during the time jump.
3. Choose **View > Attach Camera** and attach the viewpoint to one of the bank tellers to observe the animation from "behind the teller's cage." Next, detach the camera and attach it to one of the customers when he first appears in the animation and follow the customer until he is disposed.
4. At any time during the run (except when Attach Camera is in use), you may change to any Named View that you previously defined or define and use some new ones.
5. You will notice that all of your 3D navigation movements are available during the animation run (except when Attach Camera is in use), so take a few minutes to experiment with these features while the animation is running.

3D Animation

You can create animation layouts in Arena 3DPlayer following one of three approaches. Each one has its merits or its limitations. It is important before you begin your work to understand these options and to determine which method best suits your needs.

Animation methods

Method 1: Starting with an Arena model and an Arena animation

One recommended method is to create your 3D animation layout *after* you have built your model (*.doe* file) and have created a 2D animation in Arena. If your model is built first, then the playback file that you generate from Arena will provide extra information, such as Arena object names, to Arena 3DPlayer. Exact matching of named constructs is important, and this is made easier with a playback file because you can select from lists of known names in Arena 3DPlayer.

In addition, with an Arena 2D animation already built, you may import the locations of the resources, paths, and other Arena constructs from the Arena animation as the basis of your 3D animation. This is a one-time file import, and you should note that 3DPlayer always imports Arena animations into a blank layout. For details on importing, see the chapter "Importing Arena Animation Information" on page 91.

Method 2: Starting with an Arena model (no animation)

This method is recommended when the Arena model (*.doe* file) has already been built but either there is no 2D animation (and no plans to create one) or you choose not to import an existing 2D Arena animation (e.g., if the animation was not drawn to scale or the perspective is unusual and you prefer to create a to-scale animation in 3D). It is still important to generate the playback file in Arena before you create your animation in Arena 3DPlayer so that the extra information, such as Arena object names, is imported into 3DPlayer.

Method 3: Starting in Arena 3DPlayer (no Arena model)

It is also possible to create the 3D animation without any link to an Arena model file. In this instance, there is no playback file from Arena before you begin work on your 3D layout. The inherent risk with this method is that you will be defining the various named constructs on the 3DPlayer side, and you must be sure to match these names manually when the model is later being created in Arena.

Because Arena 3DPlayer is driven by an Arena playback file, the names of the animation constructs should be restricted by the naming conventions of Arena. The following restrictions apply: Symbol names can be alphanumeric strings containing letters, numbers, embedded spaces, and these special characters: @, _, %, ?, #, and <period>. Symbol names may not contain any other punctuation characters such as <comma> or <colon> and they must not be reserved words such as COUNT.

SHARING THE PLAYBACK FILE

If you are working simultaneously with an Arena model and an Arena 3DPlayer animation and are making changes to the model while viewing and re-viewing the animation based on newly generated playback files, then you will need to get in the habit of first closing your playback file in Arena 3DPlayer (**File > Close Playback**) each time you are ready to make an Arena run. Your Arena 3DPlayer program must be in the Layout Editor for the **File > Close Playback** option to be available. If you don't close the playback and you try to re-run the Arena model, Arena will see that the playback file is open and will not write the new playback file (an Arena message tells you this before Arena begins to run).

Working with the Layout Editor

In this section, we'll examine the features of the Layout Editor where you create the animation layout file and assign shapes to animation objects.

Arena 3DPlayer always open in the Layout Editor (unless you are in Demo mode, which opens in and only supports Run mode). The Layout Editor is where you can create an animated representation of each Arena object, as well as some objects that have no Arena equivalents.

The Layout Editor has four functional displays that appear in three main display panes.

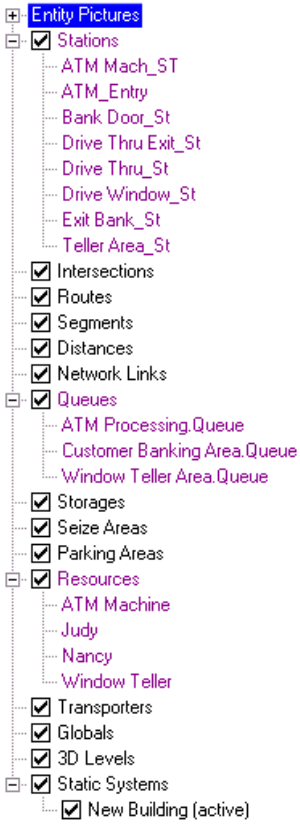
- The *Tree View pane* is located at the far left. It contains a list of all categories of constructs that can exist in the simulation and/or the animation.
- The *3D pane* is located at the upper right. It contains the 3D animation layout.
- The *Dashboard* is located at the lower right. It contains components of the animation layout that are informational rather than geometric. This may include legends, clocks, data displays, etc. Think of a flight simulator program or computer game that has a windshield or action view (the 3D pane) and a data display console (the Dashboard). The Dashboard display can be hidden by pressing the **Toggle Split Screen** control ( ) on the main window toolbar, or by selecting **View > Split Screen**.
- For some 3D-editing operations, the lower-right pane is automatically and temporarily displayed as an *Edit* pane when certain constructs are selected. At the time the

construct is selected, the Dashboard display is temporarily replaced. (The Edit pane is visible only if the Split Screen control has not been switched off.)

Let's take a closer look at these four display regions.

The Tree View pane

The Tree View pane provides your road map to the organization of your simulation and animation. It shows a list of objects that have been defined in the simulation (Arena 3DPlayer obtains this list from the playback file) and animation objects defined in the layout file. An example of the Tree View pane from the Banking Transactions model is shown below. You'll see that it lists the various components of the simulation and provides the exact names of objects from the Arena model.



The Tree View pane provides:

- a list of objects that have been defined in the simulation (i.e., are referenced in the playback file) and the layout file,
- visual cues as to which simulation objects have also been defined in the animation,
- control over the visibility of each category using the check boxes at the top level of the tree (except for entity pictures, which are visible one at a time),
- an alternate means (and in some cases, such as entity pictures and static systems, the only means) of selecting a specific object for editing, and
- a way to locate visually a specific named item in a crowded layout.

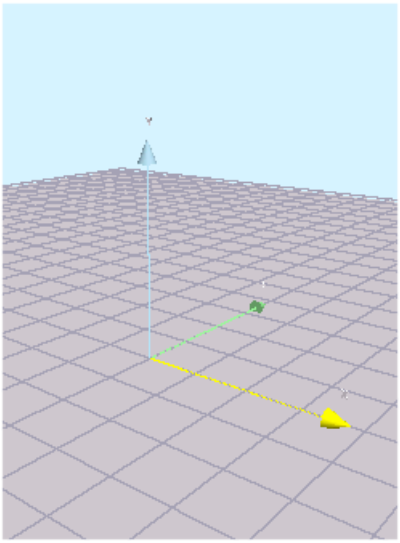
The "visual cues" mentioned above identify objects by color code. Items that exist only in the simulation playback are displayed in red. Items that exist only in the layout are displayed in blue. Things that exist in both the simulation and the animation are displayed in purple.

If a category (such as Queues) contains a mixture of red and purple items, the category name will be displayed in red so that if its view is collapsed (meaning the plus sign is visible, as in , and you don't see the individual queue names), you still know the category contains at least some things (individual queues) that have not been defined in the animation.

Category names are all displayed in the Tree View pane, even for categories that would contain objects with no simulation equivalents. (These include globals, routes, static systems, seize areas, and parking areas.) The reason all the categories are always listed is so that you can check or clear the "category visible" box.

The 3D pane

The 3D animation is created in the 3D pane of the Layout Editor. This pane contains a small number of visual elements, as shown below, in addition to the actual animation layout components.



- The *Grid* is a network of orthogonal (right-angled) lines provided for visual convenience. The snap-to-grid coordinates can align with the visual grid, but this is not required. The grid is drawn at 0 height (in the $Y = 0$ plane) and can be turned on or off. You can also control the extent of the grid and the spacing between grid lines (described in “Using the 3D grid” on page 37).
- The *Floor* is a thin, light gray box drawn just below 0 height. Its horizontal extents match the grid. This display may also be turned on or off.
- The *Origin Triad*, displayed in yellow, green, and blue whenever nothing is selected for editing, indicates the origin where $X = 0$, $Y = 0$, and $Z = 0$. It is useful as a reference point, particularly in layouts that are not centered. When you select any object, you will notice that the object’s local triad is displayed in white and is proportional to the size of the object.

The Edit pane

The Edit pane displays object-specific information pertaining to translation, rotation, scale, and color on the Transformation and Color tabs.

When an object is selected, this pane automatically appears in place of the Dashboard view (assuming that the **Toggle Split Screen** button is pressed, which it is by default). If the Edit pane is displayed and then the current object is deselected (by clicking somewhere else or pressing the **ESC** key) and no other object has been selected, the Dashboard will return automatically.

The Edit pane is described in more detail in "Manipulating 3D shapes" on page 33.

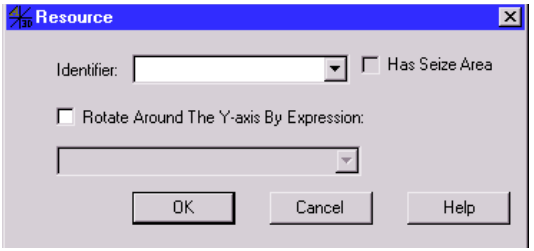
Adding 3D objects

This section explores how to add 3D objects in the Layout Editor, including resources, seize areas, queues, globals, storages, and 3D levels. Material-handling features in 3D are covered in a later chapter, as are Dashboard (2D) objects.

Resources

Resources are used in the animation to animate a comparable resource in the simulation. The shape of a resource can change according to its state, and it can have a seize area for displaying entities that are using the resource. The shapes for each state are assigned using the Shape Manager.

When adding a resource, choose **Tools > Add 3D Objects > Resource** or press the **Resource** button ( ) to open the Resource dialog.



If a playback file is open, you can press the arrow in the Identifier box to select from the list of resources that have been defined in the playback but are not yet represented in the layout. (These are also the resources that are listed in red in the Tree View pane, meaning they need to be placed in the layout.) Or you can type in any name.

The status indicator Has Seize Area (an unavailable check box) tells you whether there is already a seize area associated with this resource. This check box may not be changed from this dialog as adding or deleting a seize area is a separate function. (Unlike Arena, Arena 3DPlayer does not support "automatic" seize areas because they would frequently end up inside the 3D resource and would be difficult to manipulate.)

Arena 3DPlayer supports the rotation of resources along the *Y*-axis by an expression. When checked, you may enter an expression from the combo box to define the rotation.

When you click **OK** to accept the name of your resource, the pointer changes to a cross hair. A click anywhere in the layout places the default busy shape for a resource (a piece of machinery) at the location where your click intersects the $Y = 0$ plane. (The busy-state graphic is used in the Layout Editor to help you orient the seize area.)

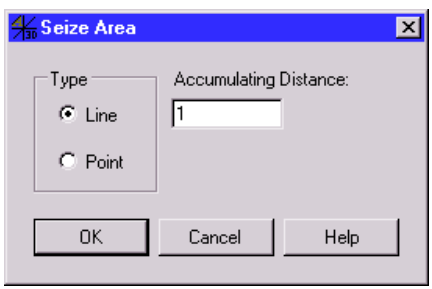
Assigning a Library Shape. You may want to use an image other than the default shape. New shapes are assigned to each resource using the Shape Manager. Begin by selecting the resource via the Tree View or clicking on the resource in the 3D pane. Then access the Shape Manager from **Tools > Shape Manager** or by right-clicking on the resource in the 3D pane and selecting **Choose Library Shape**. You will often want to assign more than one shape to a resource so that you can represent each of the resource states. Further detail on using the Shape Manager is provided in Chapter 4.

In the Layout Editor, you can view associated resource states by right-clicking on a resource, clicking the (**Resource name**) on the shortcut menu, and clicking one of the state names. This is a view-only display.

SEIZE AREAS

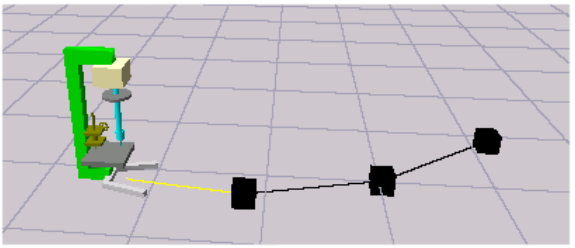
A *Seize Area* is used to display the location(s) and orientation(s) of entities that are using resources, and each one is associated with a specific resource. If there is no seize area identified for a resource, then entities using that resource will not be displayed.

Choosing **Tools > Add 3D Objects > Seize Area** or pressing the **Seize Area** button ( ) in the 3D toolbar opens the Seize Area dialog box, shown below.



The Seize Area dialog lets you specify whether the seize area type is Line or Point. If Line is selected, you can also specify the Accumulating Distance, which is the fixed linear distance between the placement of entities or transporters on a given residence-type path (queue, seize area, parking area, or storage).

After you click **OK**, the pointer changes to a cross hair, and you then click on the graphic for any resource that does not yet have a seize area. This establishes the association, and a white line appears on the screen. This line moves with the mouse, so your next click establishes the front endpoint of the seize area. You can add points by clicking repeatedly. When you are finished, you can double-click to add the last point or press the **ESC** key. The display then shows a yellow line connecting the front endpoint to the resource, a black marker where the entity using this resource will reside, and (optionally) more black markers connected by black lines (useful only if the resource has a capacity greater than one). A line-type seize area may consist of one or more points (in addition to the association point).

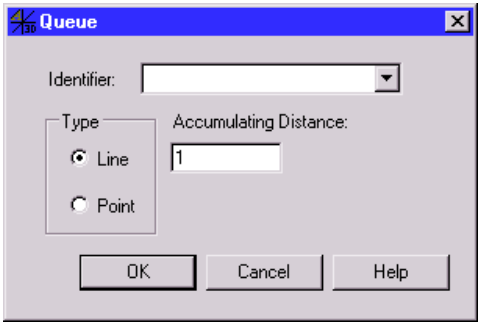


For more information on drawing and editing seize areas, see Chapter 7, "Path Drawing and Editing."

Queues

You use a *Queue* to animate a queue in the simulation. An animated queue displays entities that are waiting for something in the simulation. As with Arena, entities appear and disappear in queues (there is no smooth movement in or out).

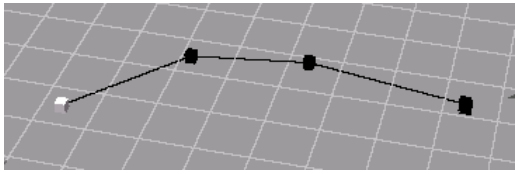
Choosing **Tools > Add 3D Objects > Queue** or pressing the **Queue** button ( ) opens the Queue dialog box, shown below.



The Queue dialog has an Identifier box for entry of the name of the queue. You can type in any name here, or if you have a playback file open, you can click the arrow to see the list of those queues that have been defined in the playback but not yet in the layout. (These are also the queues that are listed in red in the Tree View pane, meaning they need to be placed in the layout.)

The Queue dialog allows you to choose the type of the queue, which can be Line or Point. If Line is selected, you can also specify the Accumulating Distance.

After you click **OK**, the pointer becomes a cross hair, and you can click anywhere in the layout to identify the beginning location of the queue drawing (at the point where your click intersects the $Y = 0$ plane). The first click establishes the endpoint at the front of the queue. You can add points by clicking repeatedly. When you are finished, you can double-click to add the last point or press the **ESC** key. The display then consists of one white point at the front, optionally connected by black lines to one or more black points. A line-type queue must have at least two points.

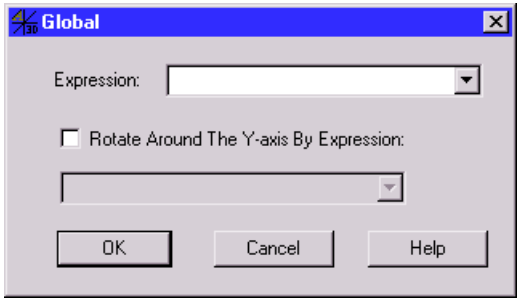


For more on drawing and editing queues, see Chapter 7.

Globals

A *Global* changes shapes in response to changes in the value of a particular status expression. This is done through trigger points. You assign the shapes and define the trigger points using the Shape Manager.

Choosing **Tools > Add 3D Objects > Global** or pressing the **Global** button ( ) opens the Global dialog box, shown below.



If a playback file is open, you can click the arrow in the Expression box and click on a status expression from a list of those defined in the playback file. You may type in any status expression, but it will eventually need to match one that is in the playback file in order for it to be animated. (It is not necessarily the case that all status expressions defined in the simulation will be in the playback. See "Status expressions" on page 104 in the "Additional Tips" appendix for more details.)

During an animation, Arena 3DPlayer supports the rotation of globals along the *Y*-axis by an expression. When checked, you may enter an expression from the combo box to define the rotation.

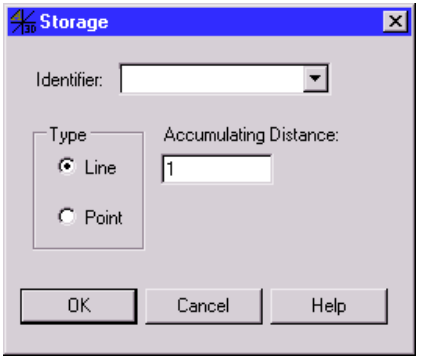
When you click **OK**, the pointer changes to a cross hair. You can then click anywhere in the layout to place the global at the location where your click intersects the $Y = 0$ plane. The shape of the default graphic for a global is a signal pole.

Since global shape animation is such that during the run the shape changes based on changing values of the expression, a different shape can be associated with the different trigger values of the expression. Each shape assigned to a given global is associated with a trigger value, meaning that when the expression reaches this value, it will cause the associated shape to appear, and that shape will continue to be used until another trigger is encountered. This eliminates the need to create a shape for each possible value of the expression. You can assign a shape for each trigger point using the Shape Manager, which is covered in Chapter 4.

Storages

A *Storage* animates a storage in the simulation. From an animation standpoint, storages behave identically to queues.

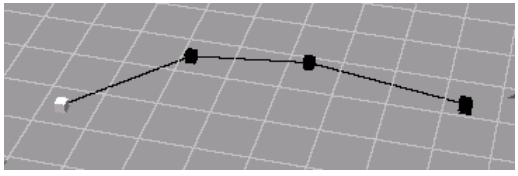
Choosing **Tools > Add 3D Objects > Storage** or pressing the **Storage** button ( ) opens the Storage dialog box.



The Identifier box permits you to name the storage. You may enter any name, or if you have a playback file open, you may click the arrow in the Identifier box to see the list of storages defined in the playback that have not yet been defined in the layout. (These are also the storages that are listed in red in the Tree View pane, meaning they need to be placed in the layout.)

The Storage dialog box allows you to select the type of the storage as either Line or Point. If Line is selected, you can also specify the Accumulating Distance.

After you click **OK**, a cross-hair pointer appears and you can click anywhere in the layout to locate the storage where the click intersects the $Y = 0$ plane. The first click establishes the endpoint at the front of the storage. You can add points by clicking repeatedly. To finish, you can double-click to add the last point or press the **ESC** key. The display then consists of one white point at the front, optionally connected by black lines to one or more black points. A line-type storage must have at least two points.

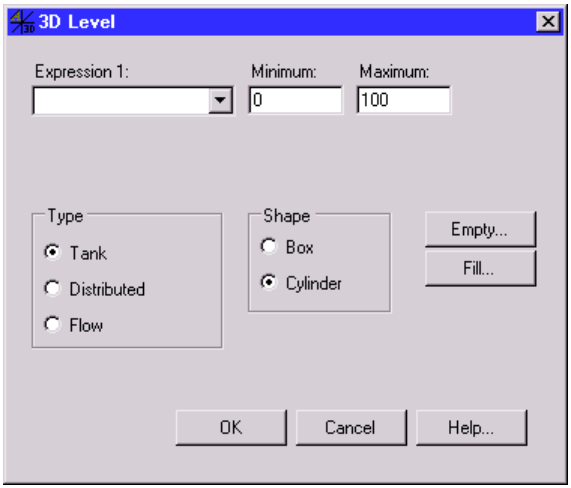


For more on drawing and editing storages, see Chapter 7.

3D Levels

Arena 3DPlayer contains the functionality to support three types of levels: tanks, distributed fill, and flow fill. Unlike all other levels in Arena 3DPlayer, these are found by pressing the 3D Level button ( ) on the 3D pane, or by clicking **Tools > Add 3D Objects > 3D Level**.

TANK LEVELS

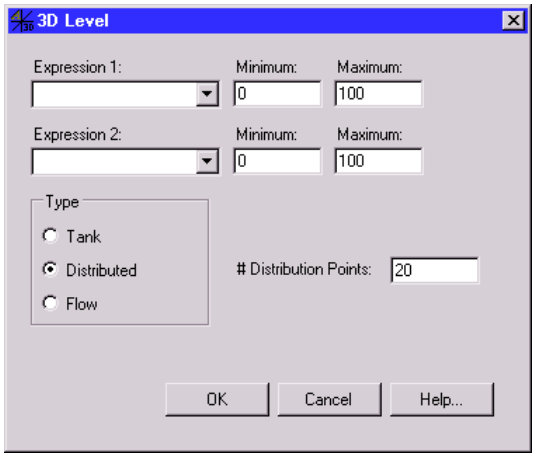


Tank levels allow for the 3D representation of the level of contents in a container. Arena 3DPlayer represents tank levels as either cylinders or boxes. Arena 3DPlayer renders the tank using the semi-transparent "Empty" color to form the outside of the tank and the "Fill" color to display the level of the contents inside of the tank. The empty and fill representations are similar to the Rectangle Level functionality of the Dashboard and Arena.

To create a tank level, press the **3D Level** button from the toolbar. In the 3D Level dialog box (**Tank** is the default setting), simply enter the Expression, Minimum, Maximum, Empty Color, and Fill Color, and choose Box or Cylinder for the shape. Once the dialog is accepted by clicking **OK**, simply click in the 3D pane to insert the tank.

The minimum and maximum value specify the range of status expression values over which the visual tank contents vary from empty to full. If the value is greater than the maximum, the tank will appear full, and if it is less than the minimum, the tank will appear empty.

DISTRIBUTED FILL LEVELS

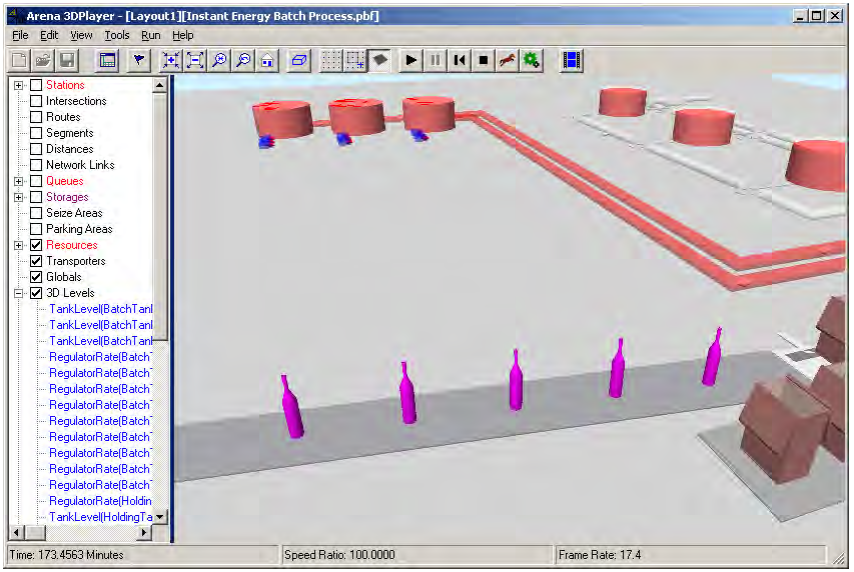


Distributed fill levels in Arena 3DPlayer fill with a discrete pattern of shapes lined up on a user-defined path. Distributed fills allow the representation of discrete but flowing items that can accumulate, such as on a packaging line conveyor. The flow is represented on the upstream section of the path, and the accumulation on the downstream section. The spacing of shapes on the downstream section is controlled by the "# Distribution Points," which is the number of shapes that would appear on a full conveyor.

Expression 1 controls the density of the flowing shapes. The higher the value is during the animation, the more shapes will be shown in the upstream section of the path, up to the maximum value specified, which corresponds to the number of shapes in the full path. Expression 2 controls the linear distance containing the stopped or accumulated shapes at the downstream end of the path. For Expression 2, the maximum value signifies a full conveyor.

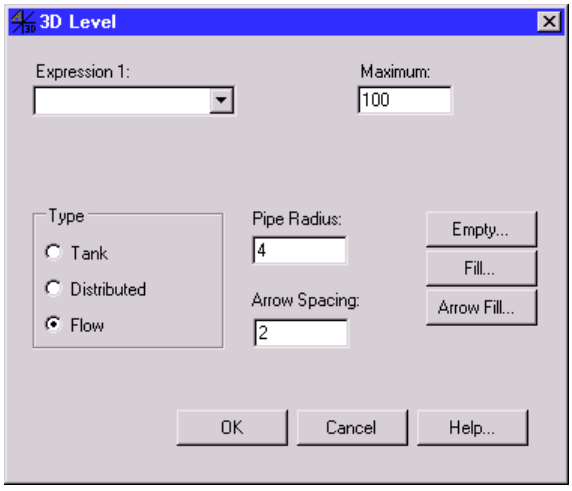
The distributed fill level can be used in any application, but may be particularly useful in animating conveyors that are modeled with equations rather than with Arena's standard conveyor constructs. Note that, if building 3D animation of a system that was modeled using Arena's Packaging template, the Packaging Conveyors can be animated using distributed fill 3D levels.

To create this level, press the **3D Level** toolbar button. From the 3D Level dialog box, you can specify one or both expressions to monitor along with the minimum and maximum for each expression. You can also specify the number of distribution points. This specifies the number of discrete points to display. If the setting of 20 is used, then each point represents 1/20 of the maximum value.



A distributed fill level is inserted by using the mouse to click a path, just as you would any normal path in Arena 3DPlayer. After you click **OK**, the pointer becomes a cross hair, and you can click anywhere in the layout to identify the beginning location of the distributed fill drawing (at the point where your click intersects the $Y = 0$ plane). The first click establishes the endpoint. You can add points by clicking repeatedly. When you are finished, you can double-click to add the last point or press the **ESC** key. The display then consists of one bottle-shaped point at the front and at the end, optionally connected by black lines to one or more black points. (Details on path drawing are included in "Path Drawing and Editing" on page 85.)

FLOW FILL LEVELS

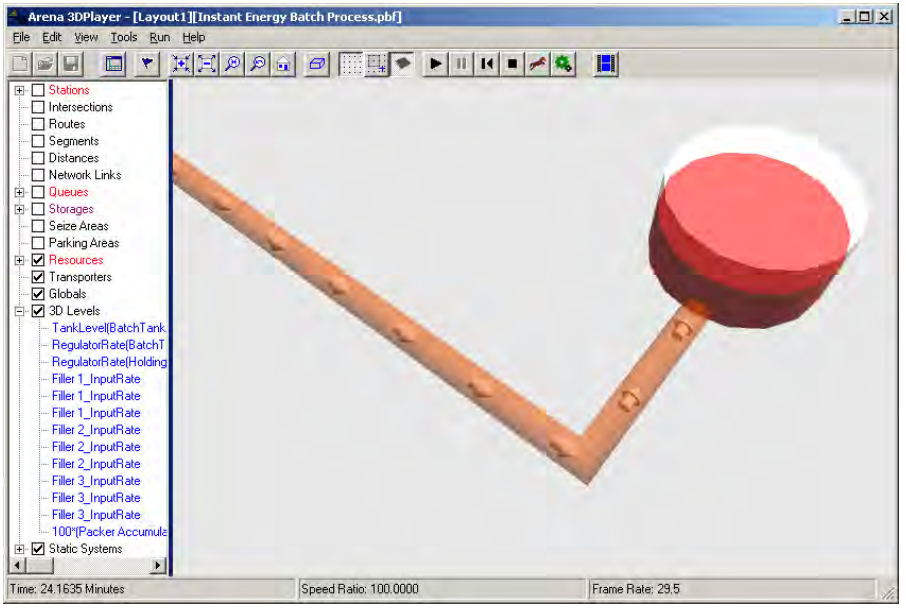


Flow fill levels allow for the representation of a flow taking place between two tank regulators. Arena 3DPlayer renders the flow fill level as a cylindrical pipe around a path.

To create a flow fill level, press the **3D Level** button on the toolbar. In the 3D Level dialog box, select the **Flow** option button. From the dialog, you can set the Expression to monitor; the Maximum; Pipe Radius; spacing between the flow arrows; and the colors for Empty, Fill, and Arrow Fill options.

After you click **OK**, the pointer becomes a cross hair, and you can click anywhere in the layout to identify the beginning location of the flow fill (at the point where your click intersects the $Y = 0$ plane). The first click establishes the endpoint at the front of the flow fill level. You can add points by clicking repeatedly. When you are finished, you can double-click to add the last point or press the **ESC** key. The display then consists of one white point at the front, optionally connected by black lines to one or more black points. A semi-transparent pipe surrounds the path, based on the radius settings in the dialog box. (Refer to “Path Drawing and Editing” on page 85 for more information on paths).

Like the tank level, Arena 3DPlayer renders the shape of the level using the semi-transparent "Empty" color. When flow is present, Arena 3DPlayer renders the shape of the level using the semi-transparent "Fill" color. Arena 3DPlayer animates the movement of the fill arrows based upon the relationship between the current value of the expression and the maximum value.



There is a maximum arrow speed for depicting the flow. (This is determined internally because too-rapid arrow movements would cause "strobing," and the flow would be hard to see or may appear too slow or backward.) You should set the Maximum property in the dialog so that the range of arrow speed changes (from stopped to the internal maximum arrow speed) matches the range of rate value changes you want to visualize. During the animation, any rate value greater than Maximum will display arrow movement at the Maximum speed.

Entity pictures

Entity pictures are used to associate an animation symbol with an entity. The symbol of the current picture associated with an entity is used to represent the entity in the animation (for example, an entity waiting in a queue, moving along a route, etc.).

The list of all entity pictures defined in an Arena model is contained in the playback file. Because of the relationship of the pictures in the model with the playback file, the Entity Picture list in the Tree View pane of Arena 3DPlayer does not appear until a valid playback file is loaded.

Arena 3DPlayer does not permit adding new pictures to the Entity Picture area of the Tree View pane. Instead, Arena 3DPlayer recognizes only those entity pictures that were created in Arena and are present in an open playback file. This includes the pre-defined shapes for the 25 Arena "built-in" entity picture names and any custom picture names (which import only as a cyan box) that were created in the Arena model.

However, once an entity picture is recognized, you can perform the full range of shape management and editing functions on it. You can display the entity shape in the Layout Editor by clicking its option button in the Tree View pane (after expanding the Entity Pictures category).

Entity picture definitions are not normally displayed while you are building or running the layout, but if you click the option button of an entity picture in the Tree View pane, a copy of it is placed near the origin (0, 0, 0) point in 3D space. This allows you to select and translate, rotate, or scale the entity picture and all entities that reference it. Normally, you will use this feature for scaling (or as a convenient shortcut to edit the current shape of that entity via the right-click menu).

Tutorial 4: Building a simple animation

Now that you've reviewed the basics of creating a 3D animation, it's time to build a simple animation using an Arena model file.

1. Start Arena and browse via **File > Open** to locate model **`tutorial4.doe`** located in the Arena 3DPlayer\Tutorials folder.

   In this simple model, customers arrive every 45 seconds on average to purchase snacks at a snack stand. They randomly choose one of two counter clerks (Clerk1 and Clerk2). Each clerk has a dedicated queue (Process 1.Queue and Process 2.Queue). The service times for Clerks 1 and 2 each follow a triangular distribution with unique min, mode, and max values. After being served, each customer exits the model. There is no "two-step" animation of this process in the Arena model.

2. You need to generate a playback file of this animation. In Arena, choose **Run > Setup > Run Control**, and check the box for **Generate Playback File**. Select the option for either **Binary Format** or **Text Format**. Be sure the destination of the **`tutorial4.pbf`** file is in a folder where you will be able to find it. Click **OK**.

   Run the simulation for a few simulated hours (the time unit is minutes, so you should run for about 200 to 400 minutes). End the simulation and close Arena (closing Arena at this stage is optional). Now you have a valid playback file.

3. Start Arena 3DPlayer, if it is not already running. By default, 3DPlayer always opens to a new, empty layout (named Layout 1).

4. Choose **File > Open Playback** and browse to the location where you saved the file **`tutorial4.pbf`** (from Step 2). The *.pbf* extension stands for playback file.

   You will see the Arena 3DPlayer Tree View in the left-hand pane. Some categories (Queues and Resources) are colored red. You'll remember that red items denote simulation items that Arena 3DPlayer sees in the playback file that have no animation equivalents in Arena 3DPlayer. A red category name means that at least some objects in that category would be red if you clicked the + symbol to expand the display.

5. Begin creating the 3D animation by placing animation objects for the two resources by pressing the **Resource** button ( ) located in the main toolbar.

   A dialog box will appear where you can define the first resource. Use the Identifier box in the Resource dialog to select **Clerk1**. (Arena 3DPlayer knows the names of the two resources that are in the simulation because they are included in the playback file.) Once the resource is named, click **OK**. Next click the cross-hair pointer once somewhere near the middle of the visible grid to place the animated resource. It is shown with the default busy shape of a machine. If the machine shape is hard to see against the gray floor, try zooming in a bit, or press the **Show/Hide Floor** button ().

   At this point, you can expand the Resources category in the Tree View pane by clicking on "Resources," and you will see that the name Clerk1 is purple, indicating that Clerk1 exists both in the simulation and in the animation. The category name remains red, because there is still a resource that has not been animated.

6. Repeat Step 5 to identify and place Clerk2. Once complete, the name **Clerk2** and the category name **Resources** in the Tree View pane will also be purple.

7. Next, place the animation objects for the two queues. Press the **Queue** button () to open the Queue dialog for your first queue. Select **Process 1.Queue** from the Identifier list. Use line queues in this tutorial. Change the Accumulating Distance to **`2`**. (Why 2 and not 0.2 or 20? If you are curious, see "Distance units" on page 105 in the "Additional Tips" appendix.)

   Click **OK** to begin drawing this queue. The queue is drawn from front to back, so place the front of the queue first by clicking a spot about 10 units (two grid lines, because the default grid line spacing is 5 units) from Clerk1. Click more points until you are reasonably satisfied with your queue. Double-click on the last point to exit the "point insertion" sub-mode.

   After you have initially completed the queue, you can select and move or delete the points or insert more points. This is covered in Chapter 7, "Path Drawing and Editing."

8. Repeat Step 7 to add the queue for **Process 2.Queue**.
9. All that remains for the layout is to add seize areas. If you do not specify seize areas, then the entities will not appear when seizing a resource.

   Add a seize area to each resource using the **Seize Area** button ( ). Stay with line-type seize areas for now and use at least two points in addition to the anchor point. Even though you expect that only one entity at a time will seize the resource, adding a segment connected by two points is the easiest way (for now) to establish the orientation of the seizing entity (later, you'll learn how to set the orientation of a point-type seize area).
10. Save the layout as **`tutorial4.a3d`**. Run the animation by pressing the **Go** button ( ).

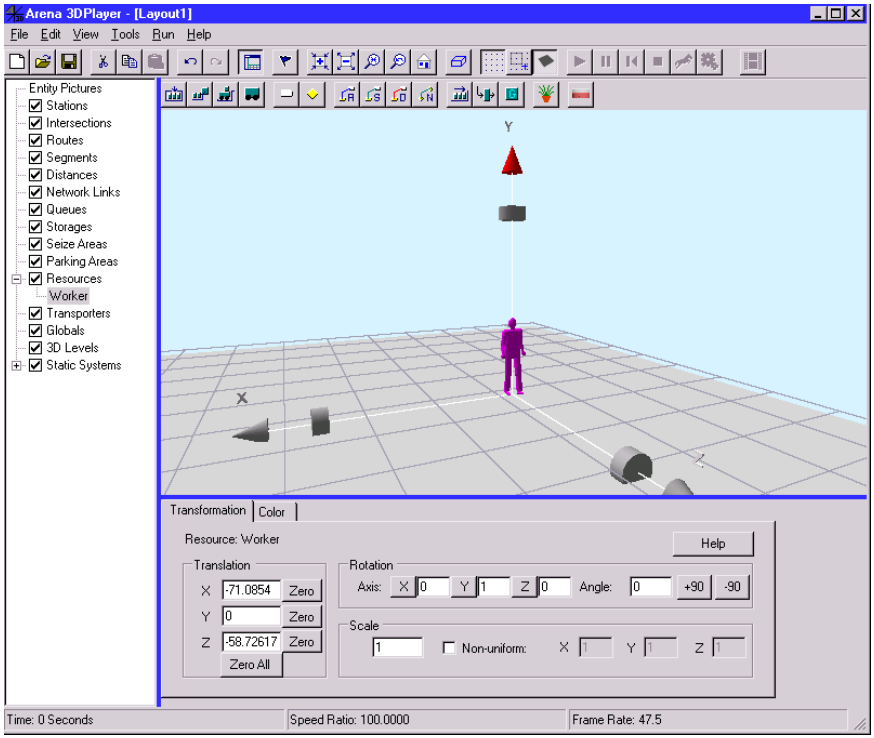
Manipulating 3D shapes

Besides assigning a shape using the Shape Manager, what can you do in the Layout Editor with a selected shape such as a resource or a global? You can:

- Cut it, copy it, or delete it
- Move it
- Scale it (in one dimension at a time, or all three)
- Rotate it (around any axis, although usually you will choose one of the coordinate axes)
- Change its color properties

In addition to the basic manipulation, you can also use Arena 3DPlayer's built-in Shape Editor to modify the contents of the shape itself. Details of Shape Editor functionality are discussed in Chapter 4.

When an animation object is selected in the Layout Editor (turns magenta), you will notice that a white "triad" or set of three coordinate axes appears. Each axis has a cone-shaped arrowhead and a cylinder-shaped collar. To move an object anywhere on the *X* or *Z* planes, simply click and drag the object. To move the object along the *Y* axis, click that arrowhead and drag the object. The arrowhead turns red while the mouse button is held down.



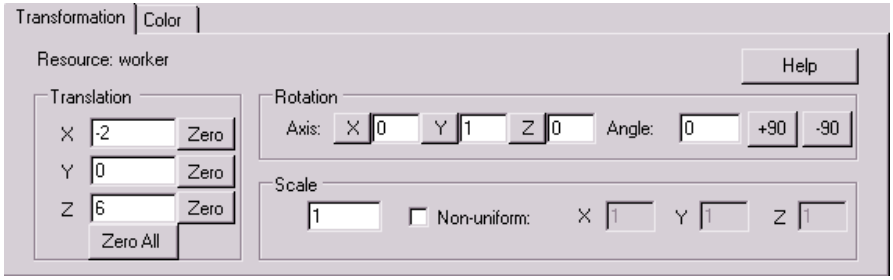
It is helpful to remember that the snap-to-grid option can make it easier to align objects along the grid or to align objects relative to one another.

The Edit pane

Precise adjustments of an object's translation, rotation, scale, and color may be made in the Layout Editor's Edit pane via the Transformation and Color tabs. To display the pane, simply select the desired object. Assuming that the **Toggle Split Screen** button is still pressed (the default), the Edit pane will appear in the lower window pane in place of the Dashboard. The Edit pane always opens with the Transformation tab on top.

THE TRANSFORMATION TAB

The *Transformation tab* (shown below) displays the exact positioning of the selected objects in the 3D pane. Object manipulations made either with the mouse or with direct keyboard entry in the Transformation tab are reflected on the tab.



To rotate an object around one of the three main axes with the mouse, you can click the collar on the desired axis of rotation (it turns red), and drag the mouse left and right. Once you have rotated the object around one axis, the graphical-rotate feature is disabled for the other axes unless you use the Transformation tab to reset the rotation to zero and/or to change the axis.

For all numeric entries in the Transformation tab, pressing the **TAB** or **ENTER** key or clicking in another field accepts the entry. Pressing the **ESC** key before accepting a field entry (or clicking on another object or on no object) closes the Transformation tab without accepting the last change.

You can use the Transformation tab instead of the mouse to perform rotation and translation. For translation, you just type a new coordinate for each dimension you want to change. For rotation, you can click on the **X**, **Y**, or **Z** button to establish the axis of rotation. *Y* is the default for a previously unrotated object, which is handy if you are turning an object around its vertical axis. You can also type in axis values to establish a vector representing a non-orthogonal axis. Once your axis is set, you can type in the absolute amount of rotation you want (in degrees), or you can use the **+90** and **-90** buttons to rotate in 90-degree increments relative to the current rotation.

There are two things to keep in mind if you rotate an object using the Transformation tab. First, be careful if you specify a non-orthogonal axis of rotation (that is, one with more than one non-zero component in its three vector components). Arena 3DPlayer will rotate around that axis, and it can be confusing. Second, rotation is “clockwise” around the

positive axis, which is a result of the Direct3D "left-handed" coordinate system (*X* pointing to the right; *Y*, up; and *Z*, away from the viewer) that Arena 3DPlayer uses.

If you set all three axis components to either 1 or -1 and rotate by 120 degrees, you can achieve the effect of rotating first around one axis and then another.

As mentioned earlier, resources and globals are shapes that you might want to manipulate in the Layout Editor. This is also true for entity pictures and transporters. Essentially, transporters work like resources and globals.

To scale an object, you can type one or more scale values in the Transformation tab. You can scale in all three dimensions simultaneously, or one dimension at a time ("non-uniform" scaling) by checking the Non-uniform box.

You can control whether different families of constructs are visible in the animation (for Layout and Run mode separately). You do this by checking or clearing the boxes in the Tree View pane. Individual shape visibility is not supported except for entity pictures in the Layout Editor. Making any entity picture the visible entity picture in the Layout Editor will allow you to manipulate its geometry.

THE COLOR TAB

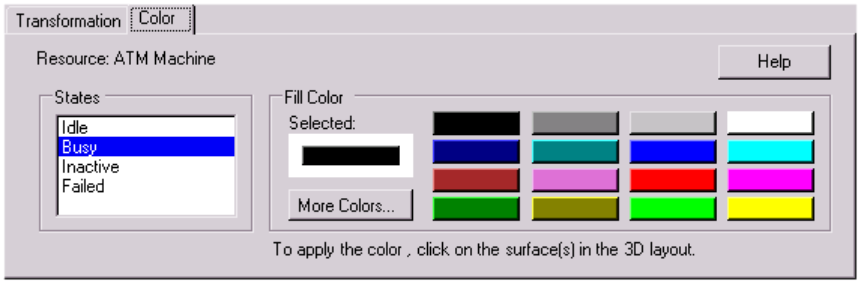
The Edit pane also contains a *Color tab* for changing the color of 3D pane objects. When clicked, an object typically turns magenta as a visual cue to identify that it is the selected or highlighted object. It remains magenta throughout the manipulations of the Transformation tab. However, when the Color tab is chosen, the selected object is displayed in its true coloration to ease the process of "click and paint."

To change the colors in an object while using the Layout Editor, select the new color from the standard 16-color palette or press the **More Colors** button to define a custom color. Custom color settings will be retained for use on subsequent invocations of this dialog. Your color choice will be displayed in the Selected block. Then simply "click and paint" the parts of the object you wish to change. To paint with another color, simply repeat the color-selection step.

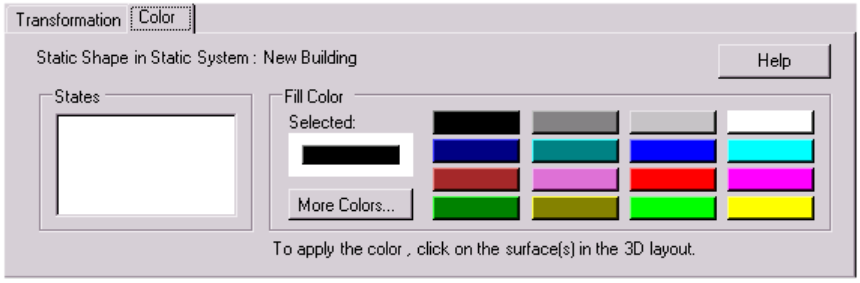
Three-dimensional objects are often comprised of many combined shapes and colors. When re-coloring an object while editing your layout, you might need to click several shape segments to achieve the desired effect, or you might need to change more than one color. (You may find it helpful to use the pan and zoom features during re-coloring.) Keep in mind that color changes applied in this manner will be saved with the layout but will not affect the shape library.

There is also a Color tab in the Edit pane of the Shape Editor (see "Shape Editor basics" on page 48). It is important to be aware that the Color tab in each area has a different method of applying color.

Resources and transporters may be represented in your animation by multiple states, and globals may have multiple trigger values. Selecting from the States or Trigger Values list of the Color tab lets you view and change the colors of each accompanying image. Each shape must be edited separately. The image below shows the display for a typical resource.



Static shapes and entity pictures (including those imported from an Arena layout) do not have states or trigger values, so the States box is empty, shown below.

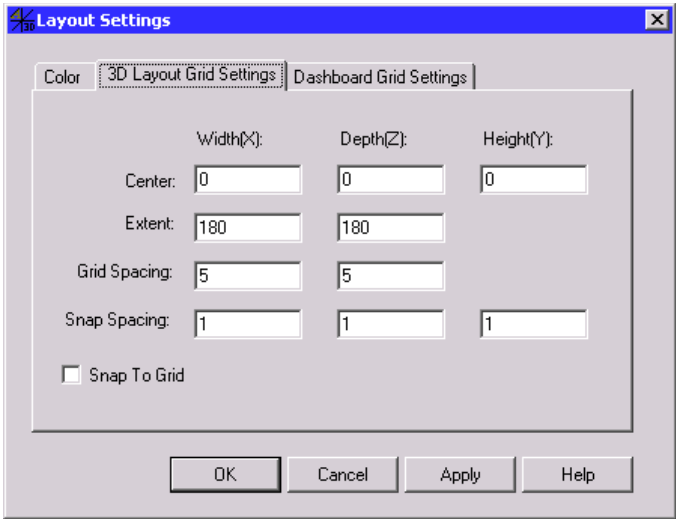


Using the 3D grid

The 3D grid is an important aid when you are editing in 3D. You can display or hide the visible grid using **View > Grid** or the **Grid** button ( ) and the visible "floor" using **View > Floor** or the **Show/Hide Floor** button ( ).

Whether the grid is visible or not, you can use it as a snap guide. You can switch **Snap to Grid** by pressing the ( ) button, whose image shows a wire frame box and cross-hair pointer. When you drag an object with snap-to-grid enabled, it will snap its local origin point to the nearest Snap Spacing intersection as you drag.

To avoid cluttering the view with grid lines, Arena 3DPlayer has separate settings for the spacing of the visible lines (Grid Spacing) and the snap points (Snap Spacing). The initial defaults are displayed in the Layout Settings dialog. To change the control settings, choose **Tools > Layout Settings** to bring up the 3D Layout Grid Settings tab, shown below.



The top two rows in the Grid Settings dialog control the location and size of both the visible 3D grid and the floor. You can change the center point (*X*, *Y*, and *Z*) and the width (*X*) and depth (*Z*). Note that if you are trying to position the grid and floor beneath some existing geometry that is not near the origin, you have the choice of either moving the grid to where the geometry is or moving the geometry closer to the origin.

Grid Spacing determines where the grid lines will be drawn. There is an important secondary purpose to the Grid Spacing and that is to determine the size of the various marker widgets that appear while you edit in 3D. These markers include seize areas, ride points, plus the four types of markers used in path drawing (point-type path arrows, line-type path node boxes, stations, and intersections). All of these are controlled by the width spacing value. So if you want bigger stations or nodes, and zooming in and out, changing the viewing angle, or switching temporarily to wire-frame mode does not solve the immediate problem, you can always increase the grid spacing in the width dimension to change the marker size.

Snap Spacing determines the spacing for 3D snap-to-grid. By default, Snap Spacing is 1 and Grid Spacing is 5, meaning that you can snap objects to four positions between a pair

of grid lines as well as to a position on a grid line. If you want to mouse-drag-translate in the vertical direction, you will find that snapping also works in the *Y* direction.

The Snap To Grid option allows you to align a selected object or objects with the snap points in the drawing area.

3D multiple selecting and editing

You can manipulate or delete multiple objects at once by multiple selecting the objects.

There are two main ways to perform multiple selection, in general (plus two more for paths described in Chapter 7, “Path Drawing and Editing”).

- One way is to hold down the **CTRL** key while clicking on the second, third, fourth, etc., object that you want to select. CTRL+clicking can also be used to remove objects one at a time from the current multiple selection.
- The other way is to hold down the **ALT** key and draw a box or “net” around the objects that you want to select. Note that this method will ignore non-static shapes such as resources if you have an active static system. For more information on this, see the section in Chapter 4 on “The active static system.”

You can also select everything in the animation at once (except static shapes, which are discussed in the next chapter) using **Edit > Select All**, or keyboard controls **CTRL+A**.

When a group is selected, you will see more than one magenta object, and you will see a triad associated with one of the objects (this is the “master” object in the description below). A multiple selection is a temporary grouping that goes away (or is deselected) as soon as you click somewhere else or press the **ESC** key.

Group Translation moves all the objects together. The numeric values shown in the Edit pane for translation are those belonging to the master object (the one with the triad showing). Changes, either from dragging the first object or from changing its numeric values, are applied equally to the other objects in the group. Remember that dragging an object by the triad cone moves the object along one plane only, which is necessary for movement along the *Y* axis. However, clicking-dragging an object itself allows simultaneous movement along both the *X* and *Z* axes.

If your multiple selection includes some but not all shapes from a rotated static system, then the translation may produce an unexpected visual result as things will not necessarily move in parallel. This is a rare case, though.

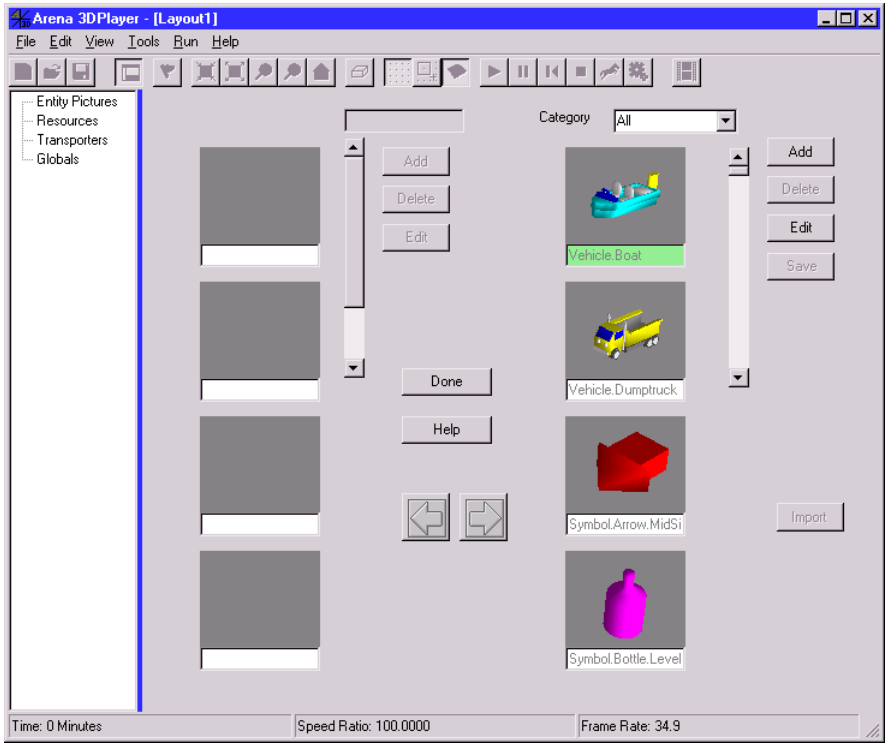
Rotating a multiple selection rotates each object individually, yet simultaneously. The shapes are not translated as a result of group rotation. The numeric values associated with the first object will overwrite any differences in the other object; that is, all the shapes will end up rotated by the same amount around the same axis.

4 Shapes and Static Systems

Working with the Shape Manager

One of the fundamental things you will do when building animations with Arena 3DPlayer is to assign shapes to animation objects. The Shape Manager provides this capability.

If you start with a blank layout and select **Tools > Shape Manager**, you'll see the following screen.



The Shape Manager displays two panes—*Tree View* and *Graphics List*. In the Tree View pane, you can see and select objects, but only those that are defined in the animation; that is, the object identifier must be purple or blue. If it is red, you can see it in the Shape Manager Tree View pane, but you can't select it there. In cases where the object is defined in the animation but not the simulation (as is always the case for *static shapes*), you can't

see it in the Shape Manager Tree View pane. However, it is possible to select any shape-assignable object in the Layout Editor and then access the Shape Manager for that object using the shortcut menu (command Choose Library Shape) that pops up when you right-click.

The Graphics List pane contains two columns of windows where 3D pictures are displayed and may be selected to represent shapes in your animation. The left column shows the current shape or set of shapes, and the right column is where you browse the library for a new shape. The Shape Group box above the left column identifies the current shape or shape set. The Category box shows the shape library names, and you may click the arrow to reveal the entire list of libraries. Once you select a library, you may use the scroll bar or keyboard arrows to view the entire set of shapes contained in that library.

The **Add** and **Delete** buttons near the left scroll bar are for adding and deleting states (a state would be a trigger point for a global or a custom state for a resource). The **Edit** button allows you to modify the individual instance of the shape in the layout via the Shape Editor (see "Shape editing" on page 47).

The **Add** button near the right column inserts a blank, new entry window to the library list that you specify. You must select a specific category from the drop-down list into which you place the new shape; entries may not be added to the *All* category. This box may hold an imported shape (see the section on "Shape libraries" and the sections about "VRML"—Virtual Reality Modeling Language, a shape import format) or, by pressing the **Edit** button, it may hold a new shape that you create in the Shape Editor.

The right-hand column **Delete** button allows you to delete user-created shapes from the libraries. You will notice that the names of user-created shapes are in black type, whereas the names of Standard Library shapes are in gray type. Standard Library shapes may not be deleted or edited. However, the **Edit** button in the right column allows you to create a copy of any Standard Library shape that may be modified and renamed as needed. This new shape will be stored in the User Library file.

The **Save** button saves all changes to the library files.

The large arrows between the columns are the **Assign** buttons, which are made available once you have selected a target and its replacement shape. You may move shapes in either direction, which may be helpful if you edit a shape in a Shape Group and later choose to add the object to one of your shape libraries. Before you can copy a shape from a Shape Group into a library list, you must first press the **Add** button in the right column to create a blank window in whatever specific library list you designate. This creates the space into which the shape may be assigned. (You may not assign a shape to replace a standard library shape.)

If you enter the Shape Manager from a new layout that has no 3D objects, you will be limited to importing, adding, or editing library shapes. Once a layout contains 3D objects, you may also perform shape editing to the Shape Group objects as well as altering the shape assignments of states or trigger points.

When you are finished working in the Shape Manager, you return to the Layout Editor by clicking **Done**.

Default shapes

Every non-path object that you place in the 3D layout (or import from an Arena layout) has a default shape or shape set. This includes colored machines for resources, a multi-colored signal light for globals, colored flatbed carts for transporters, and a potted plant for static shapes. Entity pictures, which you do not explicitly place in the layout, also have default shapes, including pre-defined standard shapes for the 25 Arena "built-in" entity picture names and a small cyan box for custom entity picture names.

Selecting the target shape

To assign a new shape to an object, you must first select the object whose shape you wish to change. You may make your selection from either inside the Shape Manager or from the Layout Editor. If you are in Shape Manager, you may choose the object through the Tree View pane. As mentioned previously, this works only if the object is defined in the layout (except for static shapes, which have no equivalent simulation object).

Or, while in the Layout Editor, simply click the object in the 3D pane. You'll see the object turn magenta to denote that it has been selected, then from the main menu, select **Tools > Shape Manager**, or right-click and select **Choose Library Shape** from the short-cut menu. This method works for objects that have been defined in the layout but not the playback (which includes all static shapes), and it also works for objects that are already defined in both places.

The right-click method of accessing the Shape Manager can also be used for entity pictures. Entity picture definitions are not normally displayed while you are building or running the layout, but if you click the option button of an entity picture in the Layout Editor Tree View pane, a copy of the entity picture is displayed near the origin (0, 0, 0) point in 3D space and is selected. This allows you to take the convenient shortcut to the Shape Manager for that entity via the shortcut menu.

Once you can see the target shape in the Shape Manager, you need to select it by clicking on the mini-window of the target shape (the selected shape's label turns green). For resources, globals, and transporters, each object can have multiple shapes and you may need to select the one shape you want to assign (for example, Busy, Idle, Failed). If there are custom resource shapes beyond the four built-in shapes, or if there are more than four global trigger values, you may need to scroll to see them all or click to **Add** new ones.

While you are in the Layout Editor, you can quickly review the states assigned to resources, transporters, and globals by selecting the object and right-clicking to display the states via the shortcut menu. This information is also available by right-clicking on the object name in the Tree View and choosing the state from the shortcut menu.

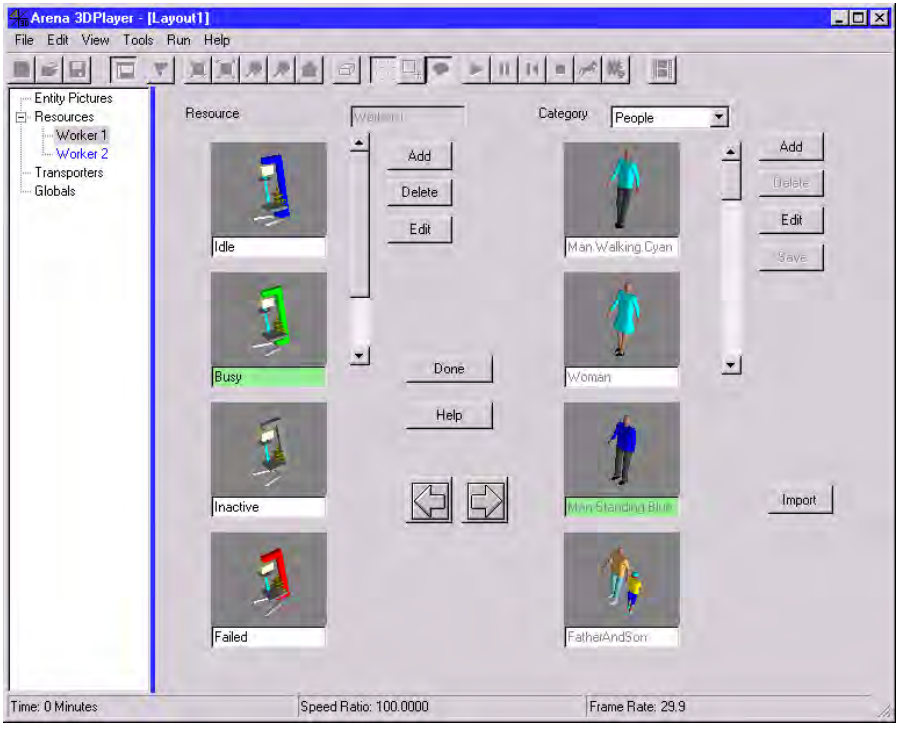
Selecting the new shape

Selecting the target is half of the assignment process. Now you need to choose the new shape from the shape library. Library shapes are organized by category, as described in more detail in "Shape libraries" on page 45. You may select a single category to browse or browse all the shapes in the right-hand column of windows. Use the scroll box to view the shapes until you find the image you want. Then click in the shape's window to highlight it (this shape label will also turn green).

Actually, selecting the target and the new shape may be performed in either order—it's purely a matter of choice.

Making the assignment

Once you have selected a new shape (on the right) and target (on the left), the selected mini-window in each column is shown with the label field highlighted in green. The picture below illustrates the appearance of the Shape Manager just before assigning a shape.



Then click the left **Assign** button ( ) to substitute the new shape for the target shape. You might need to make several shape assignments to represent all of the needed states of a resource or a transporter.

Duplicate shapes

If your animation will include multiple complex static shapes that are identical, then it is suggested that you select the shape in the Layout Editor and use the copy and paste function to place duplicate shapes rather than assigning a shape to each instance from the Shape Library. The copy-paste operation for static shapes contains a memory optimization recognized by the software that creates internal “references” to a common shape definition. This requires much less file space and video memory than assigning a shape to each one from the Shape Library.

Shape libraries

Arena 3DPlayer comes with a growing collection of pre-defined shapes in its standard library. They are stored in the following named categories:

- Vehicle
- People
- Shape (boxes, balls, cones, etc.)
- Building (internal components such as wall sections as well as whole buildings)
- Equipment
- Symbol (signage, arrows, etc.)
- Furniture
- Nature (trees, plants, shrubbery, etc.)
- Misc (shapes that don’t fit another category)

As an example, you might see a shape named Medical.Bed in the Furniture category to represent a hospital bed.

Models built using the Entity module from the Basic Process panel contain a set of default entity pictures that are loaded into the model. In Arena, these predefined names are associated with a set of predefined pictures. Arena 3DPlayer also understands this default set of 25 entity pictures names. An initial mapping between the Basic Process entity picture names and the corresponding standard shape is fixed. However, for a given layout

file, you can assign another shape to one of these built-in entity picture names in Arena 3DPlayer's Shape Manager.

It is better to create your own entity picture name in Arena and to match that in Arena 3DPlayer rather than recycle one of the 25 built-in Arena names. That way, you know in the model and in the animation what the picture is really supposed to represent.

Over time, the shape libraries will continue to be enhanced and expanded. If you create your own shapes or find shapes in the public domain that would add value to one of the libraries, we encourage you to contact Rockwell Software. Likewise, we solicit your input on other shapes you would like to see us add.

Using shapes from outside the standard library

You can add shapes to the shape library by clicking the rightmost **Add** button. These added shapes might be used in your animation anywhere you use a standard library shape.

When you click **Add** (to Library), you are first prompted to choose a category into which the shape will be placed. You must add and delete shapes from a specific category, not *All*.

Once the shape entry is added (default name "Root"), you may click to **Import** a VRML file (Virtual Reality Modeling Language—a file format for 3D scenes) or you may click **Edit** to create your own shape via the Shape Editor—a method described in "Shape Editor basics" on page 48.

To simplify the Import process, you may first select the library into which the shape will be placed. This activates the **Import** button and your VRML file will automatically be imported into the designated library.

Library changes are saved separately from the layout. The names of added shapes are represented by black text in the label, and you may change the name (without using spaces) by editing in the label's box. Standard shapes have gray-text labels, and you cannot rename or delete them.

If you modify the library in any way, you will be prompted to save your library changes when you exit the Shape Manager.

Behind the scenes, Arena 3DPlayer maintains two library files. The standard shapes are stored in a file called *standard.slb* (all categories in one file, unlike Arena), which Arena 3DPlayer never overwrites; the added shapes are all stored in a file called *user.slb*. This allows you to receive new versions of *standard.slb* without disrupting your custom shapes. It also allows you to share your custom shapes with a co-worker who may be developing similar or related animations. You should never modify the contents of *standard.slb* (in fact, you can't do so from within Arena 3DPlayer).

You don't need to transmit *user.slb* to someone who is only going to view your animation. Arena 3DPlayer stores every shape you have used in that model's *.a3d* layout file.

Tutorial 5: Enhancing the simple animation

Now that you've learned how to replace the shapes with ones that are a better representation of the animated process, you can enhance the animation from the last tutorial.

1. Re-open **`tutorial4.a3d`** that was created in Chapter 3.
2. Once the file is open, click **Tools > Shape Manager**.

   The **`tutorial4.doe`** model, from the last chapter, includes 25 built-in entity picture names that Arena placed in the playback file by default (this is discussed in the following section), plus a 26th entity picture, named **Customer2**. Customer2 is the only one used in the **`tutorial4.doe`** model, and that is the one for which you want to assign a shape.
3. Expand the Entity Picture entry in the Tree View, and click on **Customer2** from the sub-list that displays. You'll see that the shape label for Customer2 is now highlighted.
4. Now choose the People category from the right-hand column and scroll through the shapes in the windows to find the picture of the walking man in the cyan-colored shirt. Then, just as in Arena's Picture Editor, click on the desired shape (it, too, will become highlighted), and click on the Assign arrow to apply the shape to the target.
5. When you are satisfied, save your layout, and run your enhanced animation.

Shape editing

Arena 3DPlayer contains a built-in Shape Editor that allows you to edit existing shapes acquired from any source or to create new shapes as you build your animation. This tool can be accessed from either the Layout Editor or from inside the Shape Manager.

While in the 3D layout window, you may select the shape you wish to edit, then choose **Tools > Edit Selected Shape** or right-click and choose **Edit Current Shape** from the shortcut menu. Remember that you can display an Entity Picture for a selection by clicking its option button in the Tree View.

If the Shape Manager is active, you may also access the Shape Editor at any time by selecting the shape you wish to edit, either from the library list on the right or from the shape list on the left, and click the respective **Edit** button.

To create a *new* shape, you must first open the Shape Manager and then choose whether to **Add** an instance of a shape (in the left column) or add a library shape (in the right column). If you are adding a shape to the library, you will be prompted to select the specific library category into which the shape will be added. Once specified, a "blank" is added to the

column (default name "Root"), which you may access by clicking **Edit** to enter the Shape Editor.

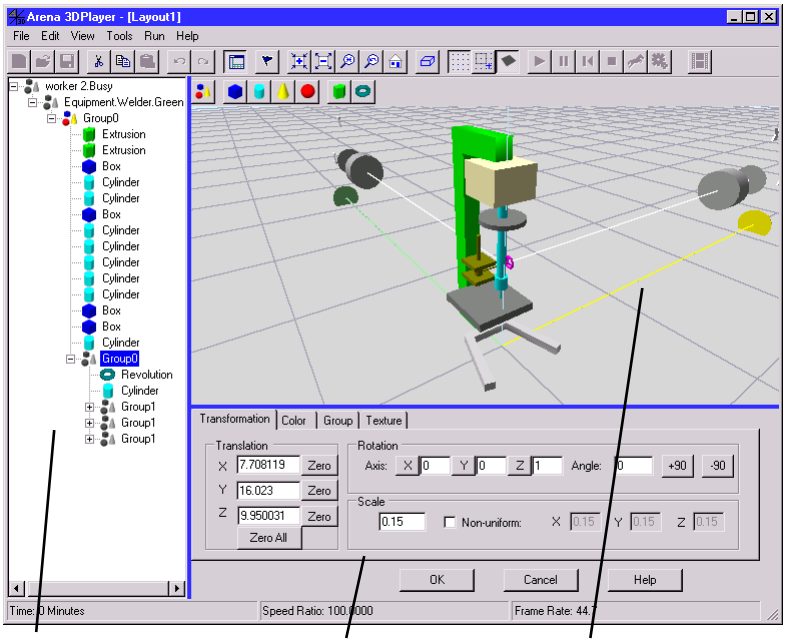
Reminders: Only shape edits that are made (and saved) as library shapes will be retained for future use. Changes made to instances of a shape will be saved for that instance only. Also remember that any shape whose name is in black type may be edited, and any black-named shape may be deleted. *Copies* of standard library shapes may be edited and renamed, but the standard shapes themselves may not be altered.

When you wish to exit the Shape Editor, click **OK** to save your changes or click **Cancel** to exit without saving. You will be returned to the screen where you invoked the Shape Editor.

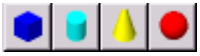
Before invoking the Shape Editor, you can choose which resource state, global trigger, or transporter state you want to edit. With the desired object selected, simply right-click the shortcut menu, click the desired resource/global/transporter name, and click the shape you wish to edit from the submenu.

Shape Editor basics

There are three regions of the Shape Editor that contain information specific to the object being edited or created, as shown below.



The main 3D workspace shows the shape being edited. You can add a new component to the shape by clicking the appropriate icon in the Shape Editor toolbar, then clicking to place it in the 3D workspace. (Step-by-step examples of drawing revolutions and extrusions using the Shape Editor are included in Appendix B “Shape Drawing: Beyond the Basics” on page 107.) The shapes that are supported include:

Commands	Description
	Add Group—a hierarchy of shape primitives
	Add Box, Add Cylinder, Add Cone, Add Sphere—basic drawing primitives
	Add Extrusion—single-segment spine. This is like a cylinder with the sides at right angles to the floor, except the cross-section can be an arbitrary polygon instead of a circle Add Revolution—circular spine and the cross-section that can be an arbitrary polygon

To the left of the 3D workspace is the Tree View pane, which displays a hierarchical listing of all components in the shape. The list will always include a group symbol (see the section on “Groups” on page 50). The plus (+) or minus (-) symbol to the left of any group symbol indicates the presence of a component list of the group, which may be collapsed or expanded as necessary. An individual shape primitive may be selected by clicking on it in the Tree View or in the 3D workspace.

The third region of the Shape Editor is the Edit pane, which contains four property tabs for attributes of the shape being edited: Transformation, Color, Primitive (Box, Cylinder, Cone, etc.), Texture.

The **Transformation** tab of the Shape Editor has the same fields as the Transformation tab of the Layout Editor’s Edit pane. It allows you to change the position, orientation, and scale of the shape.

Be aware that an origin triad and a shape triad exist in both the Shape Editor and in the Layout Editor and that movement away from the origin triad in the Shape Editor might yield unexpected results.

The **Color** tab has a different color-changing paradigm from that in the Layout Editor. In the Shape Editor, you first select the geometry (which may be any primitive or combination of primitives), and then you choose a paint color.

The (**Primitive**) tab in the Shape Editor identifies the type of primitive that is selected and will allow changes to the dimensions of the drawing shapes, but will not permit dimensional changes to the surfaces built from triangles (also known as Indexed Face Sets in the VRML world).

Using the **Texture** tab of the Shape Editor, you can apply a bitmap image to certain primitives in Arena 3DPlayer. To do this, you select the Texture tab and browse to the image file you want to use. Formats supported include *.bmp*, *.gif*, and *.jpg*.

Once you have chosen an image file, it will be displayed in a small window on the Texture tab. At this point, you can click in the 3D workspace on the surface to which you want the texture applied. This surface can be any single side of a box; the side, top, or bottom of a cylinder; the side of an extrusion; a revolution; or the side or bottom of a cone. The bitmap image will be stretched as needed to cover the surface.

Textures are useful for many things. For example, you can use scanned images to create a life-like representation of a boxed product. You can put names or other information on the side of a tractor-trailer, or you can make "signage" to display in any part of your 3D animation.

Once added to an object, a texture surface can be removed only through the Texture tab of the Shape Editor. To remove a texture, click the **Clear Image** button to delete the image from the Texture window and then select the portion of the object displaying the texture you wish to remove. The object surface will revert to the last color that was applied.

Keep in mind, however, that texture mapping can slow down the animation. Be sure you have modern graphics and computer hardware if your animations will include a lot of textures. You may also be able to adjust the Direct3D display settings of your graphics adapter to favor "speed" over "quality" of texture rendering.

GROUPS

All of the primitives described above can be arranged in hierarchies known as **Groups**. Every shape has at least one group at the "root" of its hierarchy. It is also possible that groups can contain other groups. For example, a human being might consist of a head group, a torso group, a legs group, and an arms group. Each leg might have a thigh part and a shin group, and the shin group might have a shin part and an ankle group, etc.

You can select, copy, paste, move, scale, change the color of, or delete an entire group all at once. You can also manipulate the group's primitives.

At any time in the Shape Editor, there is one group that is selected as the **Working Group**. As you click through the Tree View, you will see the change in the icon color of the Group icon. The working group is always denoted by the multi-colored Group icon in the Tree View and indicates that it is the group into which any newly created shapes will be placed. All other groups in the Tree View display a black-and-white version of the Group icon.

Adding and organizing components and groups

Shape primitives or groups that are added into the Shape Editor are also placed into the Tree View pane's hierarchy. By first choosing the desired hierarchy level in the Tree View, consideration can be given to shape location in the hierarchy before new shapes are added. New shapes can be added to the existing hierarchy structure at the chosen level. At any time, however, you can move primitives from one group to another or move a group to another tier in the hierarchy or even move a group into another group. Simply drag the primitive or group within the Shape Editor's Tree View pane to organize a structure that is most appropriate to your needs.

With practice, you'll find that this is a powerful technique for organizing your components and groups for copying, moving, and deleting them. When you change a primitive's parent group, the absolute transformation doesn't change (that is, it doesn't move on the screen). Instead, the transformation values relative to the new parent group change automatically as needed.

ADDING PRIMITIVES

Primitive components, such as a box, sphere, cone, or cylinder, are added via the Shape Editor by first clicking on the appropriate toolbar button. Place the primitive by clicking the cross-hair pointer at the location where you want the object. You can change the location of the primitive in the 3D workspace by dragging the selected item with your mouse or editing the coordinates on the Transformation tab of the Edit pane. **Zero All** places the object at the zero point of its parent group.

Arena 3DPlayer recognizes three other primitive types: extrusions, revolutions, and indexed face sets. For detailed examples of drawing extrusions and revolutions, see Appendix B "Shape Drawing: Beyond the Basics" on page 107. Indexed face sets, which are collections of triangles that can represent just about any complex 3D shape, cannot be added or edited, although they can be copied, moved, reorganized, and deleted. If you see an indexed face set ( ) in your shape's tree (as is the case for some library shapes, as well as many shapes that you might import via VRML), it means that the shape was originally created using some other program.

ADDING GROUPS

To make a new group, press the **Add Group** button. The pointer changes to a cross hair and you can place your group anywhere on the screen. Often you will then press **Zero All** to fix the initial location of the group within the framework of its parent group.

About VRML files

Arena 3DPlayer supports the text-based VRML97 file format for 3D shape graphics. VRML is supported as an export format by most 3D graphics programs, and VRML97 (a slight variation of VRML 2.0) is the preferred standard format.

VRML files are typically designated with a *.wrl* extension, which stands for "world." As that name implies, VRML is a rich file format that can support many primitives beyond the static shapes that Arena 3DPlayer needs.

To import a VRML shape into the library, choose the rightmost **Add** button in the Shape Manager, and then click **Import**. Arena 3DPlayer will prompt you to locate a *.wrl* file, and then will try to import that file. Several factors (detailed in the next section) will determine whether it will import successfully. If it does succeed, you will see the shape.

Which VRML files will import?

VRML is a file format for describing 3D scenes, with many options outside the scope of being a repository of individual static shape definitions, including built-in animations, sound files, audio clips, lighting effects, and other features. The following VRML Nodes are supported by Arena 3DPlayer:

- Appearance
- Box
- Cone
- Cylinder
- Group
- IndexedFaceSet
- IndexedLineSet
- Material
- Normal
- Shape
- Sphere
- Transform

Not every *.wrl* file will import completely, and some will not import at all.

Files ending in *.wrl* might represent either of two incompatible VRML versions (1.0 vs. 2.0/VRML97) and might be either text or compressed binary. Arena 3DPlayer is looking for VRML97 (or 2.0) and text formatted files only. If the VRML file is the wrong version, or was saved in compressed binary, it will not be imported. You may be able to overcome these difficulties by re-saving or re-exporting the file from its source application.

If you try to read in a VRML file that contains an unsupported shape node, such as an Extrusion, you will see a warning that not all parts of the graphic will be included. If necessary, you may be able to get those parts included by re-saving or re-exporting the file from its source application and choosing an option to convert Extrusions to IndexedFaceSets (IFS).

VRML features that are not supported by Arena 3DPlayer are ignored. Texture map references in a VRML file are also ignored.

VRML is a file format with an irregular structure. If the VRML file contains certain VRML primitives that could otherwise be ignored but cannot be parsed because they are unsupported in Arena 3DPlayer, Arena 3DPlayer might not be able to import that file at all.

Other drawing tools for shapes

Arena 3DPlayer's integrated Shape Editor should meet most of your shape-editing needs. However, if you need a more sophisticated 3D drawing tool, there are many standalone options from which to choose. Standalone tools include options ranging from high-end 3D creation programs such as 3ds Max®, Maya®, Pro/ENGINEER®, and others to entry-level tools like Vizx3D, that may be easier to learn, inexpensive, and suitable for making the types of shapes you can use in your animations. Rockwell Software does not endorse any of these external tools, nor has Rockwell Software tested all of them with Arena 3DPlayer. Before committing resources to any standalone 3D drawing tool, make sure that you can create and export VRML shapes using that tool and that they will import successfully into Arena 3DPlayer.

If you use any drawing program to export a shape in VRML, be sure you set any available options to export complex shapes as IFS, to export IFS as triangles, and to save the file without compressing it.

Static systems

Your animations will often contain 3D information that is important to the 3D visual scene but is not part of the simulation animation. Examples include:

- Facility layout details
- Walls and doors
- Windows and floors
- Furniture and fixtures
- Signage
- Roadways and landscaping
- Bystanders
- Indoor decorating

Arena 3DPlayer manages this information in static systems, which are listed in the Tree View. You can have zero, one, or multiple static systems in an animation, and can control the on/off visibility of each one separately. Static shapes are added in several ways:

- Import of 2D or 3D line drawings from a CAD or similar program
- Import of 3D drawings from a 3D drawing program
- Import of 2D shape graphics from an Arena model
- Addition and edit of static components directly from within Arena 3DPlayer (using built-in shapes and/or shapes found outside Arena 3DPlayer)

Creating and populating a static system

You can create a new static system by choosing:

- **File > Import DXF**
- **File > Import Arena Layout** (covered in Chapter 5)
- **Tools > Add Static System**

The Import features automatically create, name, and populate a new static system each time they are used. If you choose the second option, remember that you can import the Arena information only once into a given Arena 3DPlayer layout.

If you choose **Add Static System**, you will be prompted for a name, and then the system will be created for you with no static shapes in it; see "Adding shapes to a static system" on page 56. (One static system, named "Default," is created automatically with a new layout.)

The features for managing static systems work the same regardless of the way the system was created.

THE ACTIVE STATIC SYSTEM

Each static system has an "active" property check box in its main dialog (you can bring up that dialog by double-clicking the static system's name in the Tree View pane) that is checked to set that named system as active. Only one system can be active at any time, and the active status may be reassigned by simply opening any other named system and checking its Active box. Or to remove active status, simply open the dialog box and clear the Active box.

The active property is not the same as the visible property (the check box in the Tree View pane). A temporary label "(active)" appends to the static system name in the Tree View pane that is checked as active.

The active static system has two important characteristics. It is the system to which shapes will be added when you select **Add Static Shape**. Also, if there is an active static system, then that system's components are the only static shapes that can be group-selected by using the mouse to region-select (i.e., by drawing a net using **ALT+drag**).

It is not possible to region-group-select static shapes at the same time with other, non-static objects such as resources (although the selection works if only one or the other type is in the region). This is because the active static system may have its own orientation, and manipulating such a mixed group could have unpredictable results. If non-static objects are what you want to multi-select in a layout that also has a lot of static shapes, then you may need to deactivate the currently active static system, if there is one.

Importing a CAD layout

Arena 3DPlayer supports direct import of line geometry from a DXF file. DXF (which stands for Drawing Interchange Format) is a mature, standard, ASCII text-based format for drawing interchange. There are actually many versions of the DXF format

corresponding with the many versions of AutoCAD®, but you should not expect problems with any version if you have a typical 2D or 3D line drawing.

To import a DXF file, choose **File > Import DXF File** to bring up a standard File Open dialog that looks for DXF files. Choose a file. After the progress meter disappears, you will see a notification of success. The DXF file's contents will appear as a new static system, and an automatic "zoom all" is performed. Note that many DXF files are built using coordinates that are far from the origin in 3D space, so the automatic zoom all may appear to place the 3DPlayer origin at one corner of the screen and compress the CAD geometry into the other corner. You can move the static system if necessary. (Although it is possible to build your entire animation far from the origin, including moving the grid and floor displays if you want to, it's better to have the origin nearby in case you need to edit entity pictures using the Transformation tab, because they appear at the origin when selected in the Layout Editor.)

A DXF import brings lines into Arena 3DPlayer. Those that are part of Blocks are properly exploded, while arcs are converted to lines. Three-dimensional lines appear correctly in Arena 3DPlayer (proper conversion between the coordinate systems is performed automatically).

Fills, hatch patterns, and line types are among the DXF constructs that are not supported. Layers are not supported, and all 3D solid constructs (3DFACE) are not supported via DXF. (Use VRML and Shape Assignment mode for importing 3D solid data.)

The entire CAD drawing comes into a single static system in Arena 3DPlayer. The static system is automatically named using the name of the DXF file being imported. To select one or more lines to edit, use the **ALT+drag** multiple selection method. You can translate, rotate, scale, cut, copy, or delete individual lines. Note that you can't change the color of a selected line(s). Be sure the imported CAD system is the active system if you want to delete some lines.

Other available options for manipulating imported CAD data include scaling, translating, and rotating the entire static system, and turning on or off the static system's visibility. To edit an entire static system, select the name of the system of interest in the Tree View. The system will be highlighted magenta and can then be modified using the Transformation tab in the Edit pane or via the object's sensor pad and local triad.

There is no current support for an Arena 3DPlayer "promote to path" feature (an automatic way to turn a collection of contiguous CAD lines into a material-handling path), but a CAD-based static graphic makes it easier for you to draw your paths in any facility-based animation. If you need a promote-to-path capability, you might also be able to take advantage of Arena's promote-to-path feature if the path geometry is 2D, and import the completed paths from Arena.

Adding shapes to a static system

You may want to build up or augment a static system using shapes added one at a time, such as potted plants, static people, or pieces of furniture. Before adding a shape to a static system, be sure the appropriate system is marked as active, following the method stated above. Then select **Tools > Add 3D Objects > Static Shape** or press the **Static Shape** button ( ), available only when a static system is activated. This action does not open a dialog; it simply changes the pointer to a placement cross hair so you may place the default potted plant shape. Then you may right-click on that box to open **Choose Library Shape** and assign another shape from the library, including VRML shapes you have imported.

If you happen to have a VRML file that represents a large static portion of a layout, you can still import that as a single static shape. The building and landscaping in the Banking example were imported this way from a single VRML file.

If you are going to add a lot of shapes, consider starting with a new, empty static system instead of adding the shapes to an existing imported or other static system. This will allow you more control in managing your collections of static shapes.

Static shapes from Arena

When you import from Arena, you get static shapes as well as animation information. Specifically, any boxes, polygons, and ellipses in the Arena layout are converted into 3D static shapes. The Arena-to-Arena 3D import process extrudes these shapes into wafer-thin three-dimensional static shapes near ground level ($Y = 0$). You can move and also "grow" these static shapes vertically, individually or by group-selecting, to get an easy 3D effect or to change which one is topmost when viewed from above.

Static text is diverted to the Dashboard where it more naturally belongs. (Text in the 3D window, for labels and signs, is not currently supported. You can create 3D text in an outside graphics program and import it as a VRML shape, as was done in the supplied layout for the Banking Transactions example.)

3D wall systems

There may be times when you want to draw 3D walls in a static system. You could do this by placing a simple box on the floor and scaling it differently in *X*, *Y*, and *Z* to make it look like a wall with the desired length. Then you could do that again and again for each additional wall section, carefully positioning each panel to avoid uneven corners. Your walls would not have doors (unless you leave gaps) or windows, but it's a start.

Or you could draw a wall system in an outside program and import it as was done in the Banking Transactions example.

However, a far superior method is available in the Arena 3DPlayer standard library that provides simple blocks (Building category) for wall systems. There are four shapes in each of three colors:

- A wall panel that is 1 unit long (*X* direction) by 8 units (*Y*) by 0.5 units thick (*Z*)
- An open doorway panel
- A window panel
- A corner post

You can stretch these panels to any length by clicking the "non-uniform" box in the Scale area of the Transformation tab and modifying the *X* scale. So if you want a 20-unit-long wall, just change the scale to *X*:20, *Y*:1, *Z*:1. You can rotate the wall if you want it to be at a different orientation (such as north-south instead of east-west). The "long" direction will still be *X*, because the scale factors are applied before the rotation.

The three panel shapes have the unique characteristic that their zero point is centered on one end. That means when you stretch the wall, one end will stay anchored. It also means that whole-unit lengths will always end on whole-unit boundaries (very useful as you are likely to have snap-to-grid turned on when building walls). This will help you put wall systems together quickly.

The corner post is 8 high by 0.5 by 0.5 and creates a smooth corner wherever two panels meet at right angles. When you are using snap-to-grid for the wall panels, this corner will snap right into place.

The doorway and window panels are not intended to be scaled, although you can do so if you need a wider opening or a wider window.

If you are building walls in a to-scale drawing that is drawn in units other than feet, such as meters or inches, this is easy to accomplish. Please see "Distance units" on page 105 of the "Additional Tips" appendix.

Like all library shapes, the wall system shapes all have a uniform 1 x 1 x 1 scale in the Transformation tab after you first assign them to a static shape. The dimensions mentioned above are internal to the shape definition. A wall panel in a metric-units layout might need its height (*Y*) and thickness (*Z*)—but not length (*X*)—adjusted in the Transformation tab down to 0.3 to give you a 2m x 1m x 0.15m section and assure consistency of units. Or you could define the wall shapes that you want using the Shape Editor and save them as custom library shapes. Once you have one wall section looking the way you want, you can copy and paste that static shape and avoid having to edit the transformation of each section.

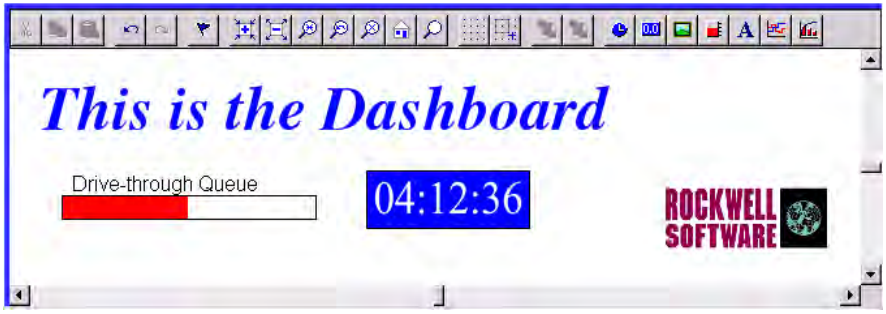
If you use an imported CAD floorplan as the template for your wall system, be sure you add the walls into their own separate static system. That way, you can easily turn the CAD floorplan's visibility on or off or even delete that system without disturbing the walls.

5 The Dashboard

The Dashboard is a 2D pane for displaying information during the animation. The display of this region is optional and is controlled by the **Toggle Split Screen** button on the toolbar of the main window.

The following types of objects may be included on the Dashboard:

- Static text
- Tables of statistics with variable text
- Clock information (analog or digital)
- Bitmapped images (such as a company logo or other decoration)
- Levels that change with changes in some expression (rectangle fills are currently supported)
- Plots and histograms



You may adjust the size of the display area for the Dashboard by moving the splitter bar that separates the 3D pane and the Dashboard. Upon re-opening a saved layout, the location of the splitter bar and the relative proportion of the Tree View pane, 3D pane, and Dashboard will be the same as when the layout was last saved. (Note: The size of the application window is retained upon exit, but is not associated with a particular layout file.)

Another feature of the Dashboard is the ability to set Grid and Snap Spacing. To modify the Grid Spacing or Snap To Grid options, choose **Tools > Layout Settings** to reveal the Dashboard Grid Settings properties tab.

Navigating in the Dashboard pane

Navigation methods in the Dashboard share some common command types with the 3D pane, but differ most notably in the use of the mouse to control movement. Horizontal and

vertical panning within the Dashboard is controlled either by the scroll box or arrow on the scroll bars or via the keyboard arrows. The commands to set the view include:

Buttons	Commands
	Named Views—opens the Named Views dialog for listing views, assigning them, editing the names, or deleting them.
	Zoom In or Out—zooms by a fixed factor, which you can change.
	View All—views all objects at once.
	View Previous—views the last zoom selection.
	Show Zoom Percentage—allows you to set the Zoom Percentage or simply to learn the current view scale.
	View Home—returns the view to the saved Home setting.
	View Region—select this button to change the pointer to a cross hair, which you then place, click, and drag the wire frame to surround the objects you wish to view in close-up.

The Dashboard supports a Home view that can be changed and saved with the layout. This is the view that will be used the next time the animation starts up. The Dashboard also supports custom Named Views that can be saved with the layout—just as in the 3D pane, but they are stored and controlled separately.

You may find it useful to become familiar with the commands available by right-clicking the mouse (the commands are different for the 3D pane and the Dashboard). The shortcut menu has the following commands:

- Undo/Redo
- Cut, Copy, Paste, Delete
- Select All
- Add Object
  - Clock, Image, Level, Static Text, Variable, Plot, Histogram
- Selected Object
  - Bring to Front, Send to Back, Change Snap Point, Properties (which opens the object dialog box)
- View
  - All, Home, Named Views, Region, Previous, Make This the Home View
  - Zoom In, Zoom Out, Zoom Multiplier, Zoom Percentage
  - Grid

Adding Dashboard objects

Objects are added to the 2D pane via one of three options. You may:

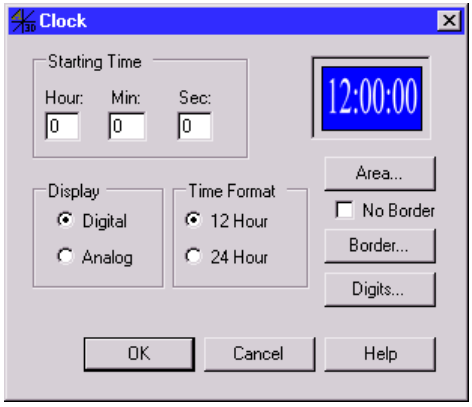
- select **Tools > Add Dashboard Object** from the menu bar
- right-click within the 2D pane and select **Add Object** from the shortcut menu
- click the desired button on the Dashboard toolbar

In addition to object placement options, the Dashboard toolbar has buttons to change the stacking order of objects, to control the view, and to control the 2D grid. You'll also notice that the Dashboard has its own grid (invisible by default).

It's time now to explore the object types supported in the Dashboard.

Clocks

As in Arena, you may also add analog or digital *Clocks* to the display (**Clock** button). The Clock dialog box identifies all properties of the clock that will show the current simulation time (in a human-readable format) while the animation is running. This dialog box is shown below.



The property settings in this dialog should be familiar to you if you have placed a clock in an Arena model. You may define the characteristics of the clock when the dialog first appears (before placing the object), or edit them by double-clicking an existing clock. Among the common setting options are background/fill area, border color, and digit/hand color.

Placement of the digital clock resembles that of the variables. You control the size of the clock by dragging the bounds of the object to the desired size. As you do so, the font size of the display increases/decreases to support the size of the rectangular shape.

The clock display in digital mode, besides being in HH:MM:SS format, is very much like a variable, with the same graphical properties of Background, Border, and Font. You may also control whether the digital clock is displayed in 12-hour or 24-hour format.

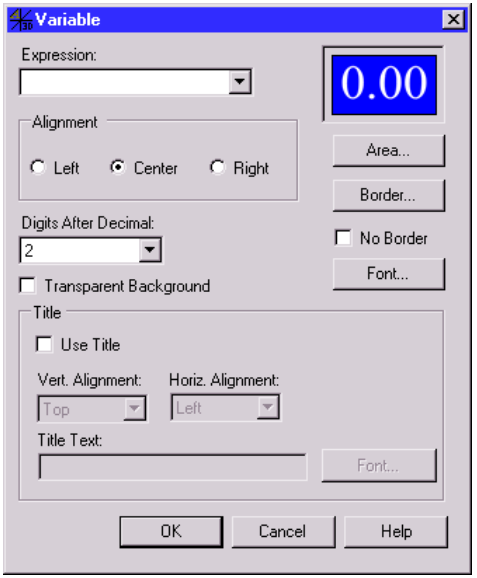
In analog mode, the clock is round with minute, second, and hour hands. The size of the analog display is first determined by the width you drag during click-drag-release. Once placed, you may change the size of the analog display by selecting the object and dragging the handles to achieve the desired size.

In either case, you may specify a starting hour, minute, and second, and the clock will equate Simulation Time 0.0 with that time during the animation.

Arena models specify their base time units in the **Replication Parameters** dialog accessed via **Run > Setup**. Note that any time delays or rates specified in a model using modules that do not define time units (such as modules from the Blocks and Elements panel) are entered as base time units. You should check to see that the Arena model has correct time units before generating the playback file.

Variables

A *Variable* is a number in the Dashboard that changes during the animation in response to changes in a status expression in the simulation. The Variable dialog box, accessed via the **Variable** button (0.0), is shown below.



If you are creating your animation using the information from an Arena playback file, you will be able to select from among the model expressions imported with that file that are displayed in the Expression list. You may type in any status expression, but it will eventually need to match one that is in the playback file in order for it to be animated. As with the Text display, you may set the Alignment and Font options of your choice. You can define the rectangular Border color and the Background color or check the box for No Border or Transparent Background. In addition, you may select the number of digits after the decimal in your variable display from a drop-down list (default is 2).

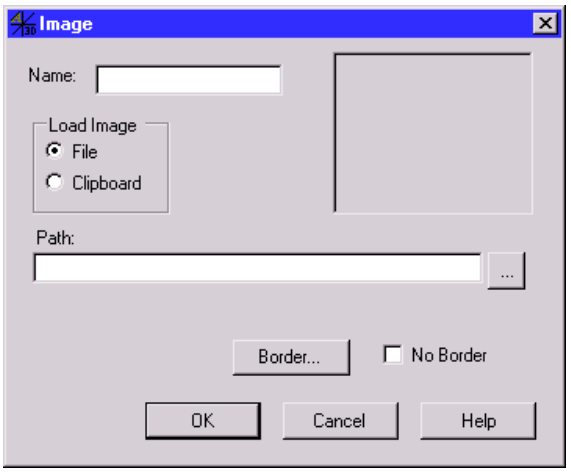
The Variables dialog also includes a titling option with alignment positioning and a text field. Once you have placed your variable, if you find that the titling isn't compatible with the size of the variable display you have set, you might need to alter the font size so the entire title is visible. Or you may prefer to clear the title box and use static text for your variable title. (The size of the box defining the variable determines what portion of the title is visible. The longer the line of text is, the larger you must drag the variable display to expose the full title.)

When your selections are made, click **OK** to close the dialog box. The pointer becomes a cross hair and you must then identify the placement of the object by the click-drag-release method. The size of the rectangle is determined by the shape you drag, but you may readjust the size by selecting the object and dragging the handles to a new position. The font size of the variable display increases/decreases as you adjust the size of the rectangle. However, the font size of the titling remains at the size you set in the dialog box.

As the simulation runs, you will be able to observe the changes displayed in the variable objects on the Dashboard. These numbers change dynamically based upon the activity during the run.

Images

Images are static 2D graphic pictures (*.bmp*, *.jpg*, or *.gif*) displayed in the Dashboard. Clicking the **Image** button ( ) opens the Image dialog, shown below.

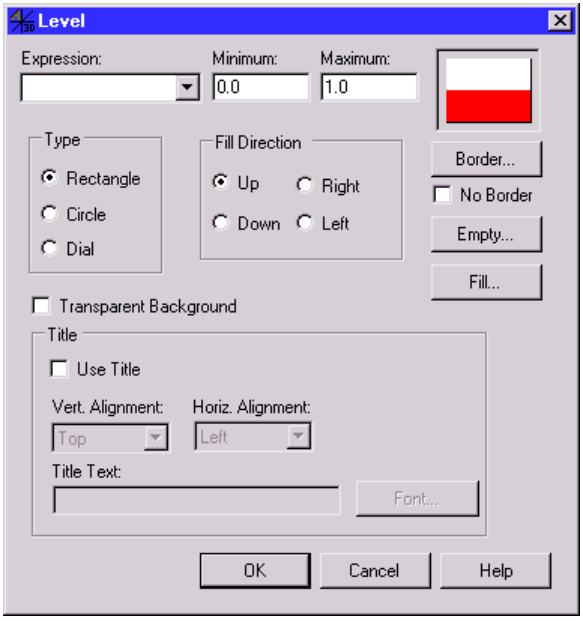


From this dialog, you can insert an image from the Clipboard or you can browse to any *.bmp*, *.jpg*, or *.gif* file via the **Path Browse** button (...). Once you have selected a file, the image will display in the preview window of the dialog box. Upon clicking **OK**, your pointer becomes a cross hair. Simply select the location where you want to place the top-left corner of the image and drag-release to complete the placement. Arena 3DPlayer forces the image to be at its correct aspect ratio, so you need only drag the wire frame to the desired width of the image. You can later scale the image using a handle, but the aspect ratio will not change.

When you save the layout file, this image, like all other Dashboard and 3D components, is embedded in your layout file. The original image file is not needed.

Levels

A *Level* is an animation plot that represents the fluctuations in the value of an expression. To add a Level plot to the Dashboard, click the **Level** button ( ).



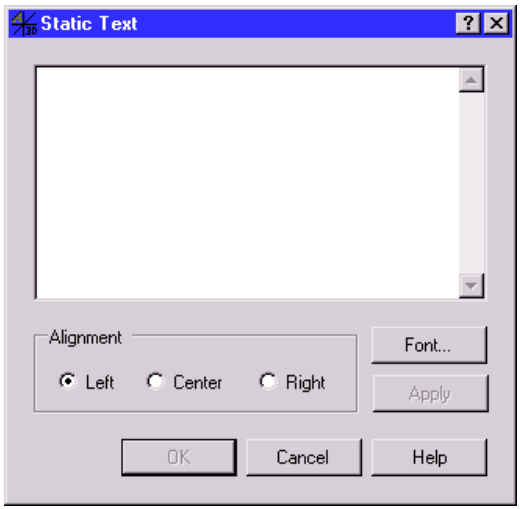
The Level dialog box allows you to specify Level type as Rectangle, Circle, or Dial; the Fill Direction; colors for the Fill and Empty portions of the level; and Border color, if any. The Minimum and Maximum boxes determine the range of values of the expression that will cause the level to be exactly empty and exactly full, respectively.

As with Variables, the Level dialog also includes a titling option with alignment positioning and a text field. Once you have placed your level, if you find that the titling isn't compatible with the size of the variable display you have chosen, you might need to alter the font size of the title so that the entire text line is visible. Or you may prefer to clear the title box and use static text for your variable title. (The size of the box defining the level determines what portion of the title is visible. The longer the line of text is, the larger you must drag the level display to expose the full title.)

When you import Rectangle Levels from Arena you can direct them to the 3D layout, the Dashboard, or both. See "Dashboard import considerations" on page 93 for details.

Text

A *Text* object in the Dashboard is a static display of text, which can be anywhere from a few characters to a long paragraph. To open the Static Text dialog box, click the **Static Text** button (A).



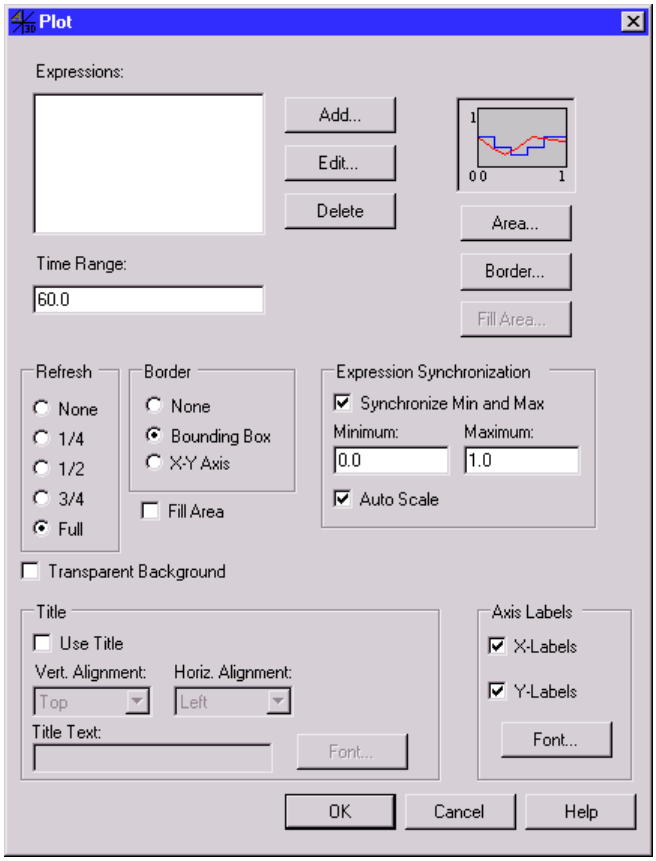
As you type, the text string will flow and wrap in the dialog. If you wish to force a line break, you must use **CTRL+ENTER** (simply pressing **ENTER** is equivalent to clicking **OK** and the box will close). Text alignment is set by selecting either **Left**, **Center**, or **Right**. The **Font** button opens a dialog box where you may set the Windows-type font properties. If you intend to share this file with others, you might want to select your fonts carefully as other computers may not have the same availability of fonts and a substitution might occur.

When you click **OK** to accept the text string, the pointer changes to a cross hair, permitting you to click and drag the wire frame boundary of your text entry.

Whether you choose the default font size (14-point type) or any other size, the text may not "fill" the insertion box that you drag. The font size display at 100% zoom factor matches the font size specified. If needed, you may may select the text and adjust the shape of the bounding outline by dragging the handles to the desired locations. Alternatively, you may double-click to re-open the Text dialog to edit the string and its alignment and font properties.

Plots

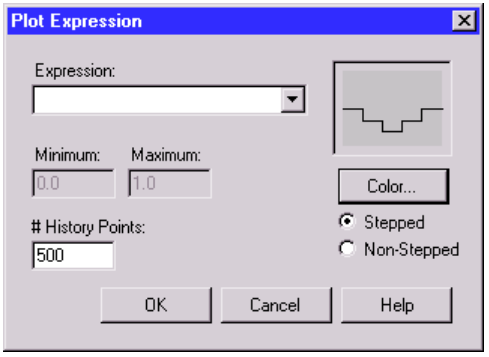
A *Plot* is is a chart or graph marking historical points of expressions monitored during the simulation run. You can use a plot to animate expressions that are defined in the playback file. To place a plot on the Dashboard, press the **Plot** button ( ) to open the Plot dialog box.



The Plot dialog box contains settings for Area, Border, and Fill Area to allow you to define the color characteristics for each one. Further options include selecting Refresh, Border, Axis Labels, and Transparent Background. The Expression Synchronization box contains check boxes to synchronize all minimum and maximum values across all expressions that are being monitored as well as a box to select Auto Scale.

Display titling may also be selected and defined. Be aware that the font settings for titling and axis labels include the ability to define the font size. One or both of these settings might need to be modified, depending on the dimension of the plot display, in order to maintain a pleasing proportion to the chart itself.

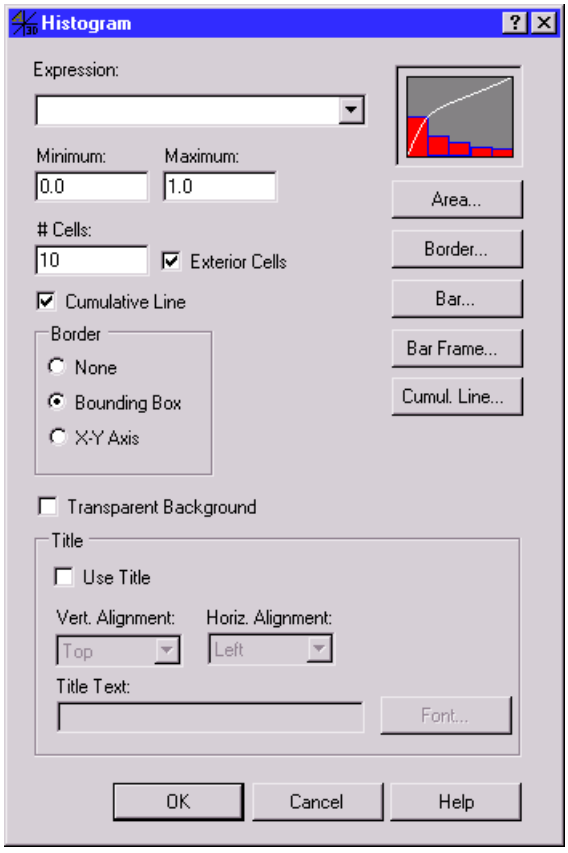
One or more expressions can be displayed on the same graph by adding and defining the expressions by means of the combo box in the Plot Expression dialog box. To add an expression, click the **Add** button in the Plot dialog, which brings up the Plot Expression dialog. If you type the expression, be sure that it matches the expression in the playback file in order for it to be animated.



The Plot Expression dialog box allows you to define the Minimum and Maximum range of values and to set the number of History Points to be stored. This dialog box also lets you select a Stepped or Non-Stepped visual display of the data points.

Histogram

A *Histogram* is a representation of a frequency distribution by means of rectangles whose widths represent class intervals and whose areas are proportional to the corresponding frequencies. Pressing the **Histogram** button ( ) opens the Histogram dialog box, shown below.



The Expression field lets you choose the expression to be monitored from the expressions listed in the combo box. These expressions were defined either in the playback file, if they were animated in the Arena *.doe* file, or from expressions explicitly stated in the Arena **Run > Setup > Run Control > Status Expressions** menu. You may type in any expression, but it will eventually need to match one that is in the playback file in order for it to be animated.

You may define the range of values and the number of cells to display or you may accept the default settings. The Exterior Cells check box option specifies whether or not observations that fall outside the range specified by the Minimum and Maximum should be shown in exterior cells. If the cell is cleared, the extra cells will not be added and outlying values will be ignored.

The buttons for **Area**, **Border**, **Bar**, **Bar Frame**, and **Cumulative Line** allow you to define the color characteristics for each one.

You may check the box to display a title with the Histogram, and you may choose the alignment positioning and title text.

More about the Dashboard

Editing objects

To select any object on the Dashboard simply click on it. You can open any Dashboard object's dialog box and change its properties by a simple double-click on the object.

If you want to alter the placement of objects, simply select the object you wish to move and drag it to its new location. To change the size, select the object and drag the handles. Dragging a corner handle resizes the object in two dimensions. For images, the aspect ratio is preserved when you drag a handle.

Other Dashboard facts

The Dashboard is unitless except for font sizes. When the zoom percentage is set to 100%, font sizes appear at their correct scale. To see or change the zoom percentage, use **View > Dashboard > Zoom Percentage**.

Using the **Bring to Front** and **Send to Back** buttons allows you to arrange the Dashboard objects so that they overlap.

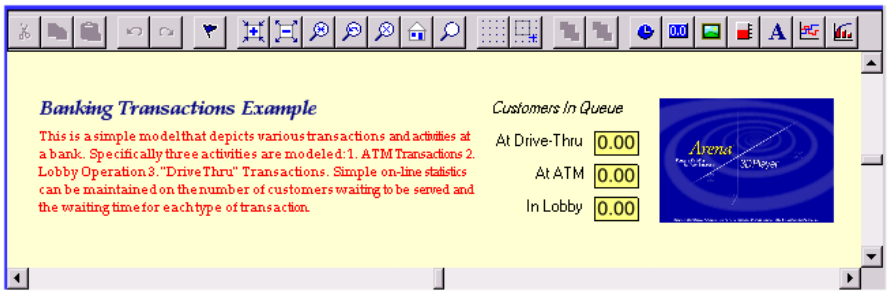
Undo and **Redo** are not multi-level and do not apply to all operations. They work to undo or redo, respectively, the most recent object insertion or deletion only.

The View commands on the Dashboard toolbar all have menu equivalents that are accessible via the **View > Dashboard** sub-menu or the right-click **View** command.

Tutorial 6: Editing the Dashboard display

If you didn't play with the various features of the Dashboard as you read through the previous descriptions, open the Banking Transactions playback file in the Examples folder.

You can experiment with moving variables and text in the Dashboard window and resizing the objects to get a pleasing display.


Banking Transactions Example
This is a simple model that depicts various transactions and activities at a bank. Specifically three activities are modeled: 1. ATM Transactions 2. Lobby Operation 3. "Drive Thru" Transactions. Simple on-line statistics can be maintained on the number of customers waiting to be served and the waiting time for each type of transaction.
Customers In Queue
At Drive-Thru 0.00
At ATM 0.00
In Lobby 0.00
Arena

1. Locate and zoom in on the main paragraph of descriptive text. Now experiment with changing the font properties. Notice how the flow adjusts when you select different font families or increase/decrease the font size. If you want to change the text boundary, drag the handles of the text box.
2. Experiment with statistics by creating a new display with variables, or add a clock and some additional static text. Don't forget about using Snap to Grid!
3. Select and set a Dashboard Home View to preserve the scale and location when you save the layout.

You can run this animation and see the Dashboard results, even if you have done nothing to enhance the 3D display.

6 Material Handling

This chapter focuses on the Arena 3DPlayer capabilities that support the built-in material-handling features of Arena. They include:

- Entities moving on Routes
- Entities moving on Segments (Conveyors)
- Transporters moving on Distances
- Transporters moving on Networks

The specification and operation of material-handling system animation constructs in Arena 3DPlayer match closely the specification and operation of the equivalent Arena constructs.

Routes

A *Route* in Arena 3DPlayer corresponds to an Arena route. Arena uses a route to move an entity from one station to another without a conveyor or a vehicle.

To enter a route in Arena 3DPlayer, select **Tools > Add 3D Objects > Route** or press the **Route** button ( ). The Route dialog will appear, showing unnamed From and To station names. The Keep Entities Vertical On Slopes check box allows you to specify that entity shapes remain vertical when they move along sloping guide paths. This is helpful to represent a person walking on a hill or a helicopter flying. Click **OK** to begin drawing your route using the path-drawing method.

After you click **OK**, the cross-hair pointer appears and you can click anywhere in the layout to locate the route where the click intersects the $Y = 0$ plane. The first click establishes the point at the front of the route. You can add points by clicking repeatedly. To finish, you can double-click to add the last point or press the **ESC** key. The display then consists of one white point at the beginning station and a white point at the end station, connected by blue lines to one or more black midpoints. You can click on an existing station for the start and/or end point; otherwise, a new station will be created. The technique is described in detail in the next chapter, "Path Drawing and Editing."

After the route has been drawn, you can change the name of the starting or ending station by double-clicking on the white shape representing the station and changing the name in the Station dialog box.

To see the Route dialog for an existing route, you double-click on the line representing any part of the route (but not on a station or midpoint). You can delete the entire route from that dialog box (the stations will remain in case they are needed by other material-handling constructs and would need to be deleted separately if they are not needed).

You may notice that the route has no identifier in the Route dialog. There is no Arena simulation construct that is equivalent to an animated route. Even if there are routes in the animation part of an Arena model, only the stations are passed as simulation information in the playback file. Because of this, there is no listing of routes in the Tree View pane, and no name matching is performed.

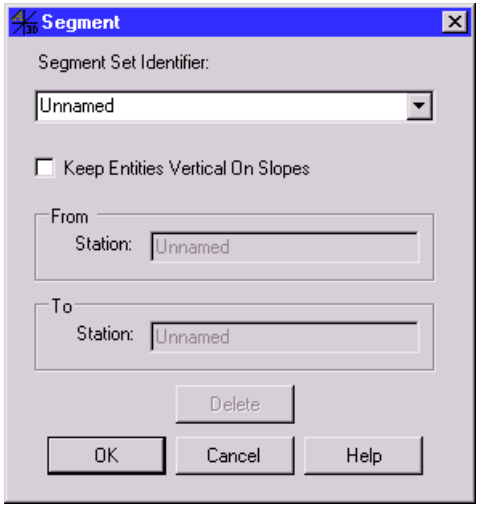
All of this behavior is consistent with Arena. Of course, if you start your layout by importing an Arena animation containing routes, those routes will be included in the graphical Arena 3DPlayer layout.

Segments

In Arena and Arena 3DPlayer, the animation equivalent of a simulated conveyor is a *Segment Set*. The segment sets and their names are present in the playback file so they can be associated with specific conveyors.

Belonging to each segment set are one or more *Segments*. Segments behave like routes in that they do not have names and each segment connects a pair of stations. Entities on conveyors in the simulation move from station to station. In the animation, these entities travel from station to station on segments.

To draw a segment, you select **Tools > Add 3D Objects > Segment** or press the **Segment** button ( ). The Segment dialog will appear, showing unnamed From and To station names. The Keep Entities Vertical On Slopes check box allows you to specify that entity shapes remain vertical when they move along sloping guide paths. This is helpful to represent a person walking on a hill or a helicopter flying.



A segment has no name, but there is an Identifier field in the Segment dialog box in which you need to identify the segment set to which this segment will belong. If you have a playback file open, the Segment Set Identifier box shows all segment sets that are in the playback.

Once you have chosen or typed in the segment set name, you then click **OK** to place the segment. The cross-hair pointer appears and you can click anywhere in the layout to locate the segment where the click intersects the $Y = 0$ plane. The first click establishes the point at the front of the segment path. You can add points by clicking repeatedly. To finish, you can double-click to add the last point or press the **ESC** key. The display then consists of one white point at the beginning station and a white point at the end station, connected by cyan lines to one or more black midpoints. (For more detailed information, refer to “Path Drawing and Editing” on page 85.) You can click on an existing station for the start or end point; otherwise, a new station will be created.

After the segment has been drawn, you can change the name of the starting and/or ending station by double-clicking the white shape representing the station to open the Station dialog box.

You can open the Segment dialog for an existing segment by double-clicking on the line within the segment's geometry (but not on a station or midpoint). From that dialog, you can change the name of the segment set identifier. You can also delete the entire segment from that dialog (the stations will remain in case they are needed by other material-handling constructs, and they would need to be deleted separately).

The Tree View displays each individual segment set, distance set, and network link when you click on the plus (+) sign for the category. If you double-click on the individual list name in the Tree View, you can identify the location of each from/to item when it "flashes" a temporary size change.

Transporters

Animated *Transporters* resemble both entities and resources. They are like entities in that they can move on paths. The paths they move on are called *Distances* and *Network Links*. Transporters are like resources in that entities compete for them and they have the ability to change their animated appearance according to their state. Entities can be placed in Transporters so that they appear and move together.

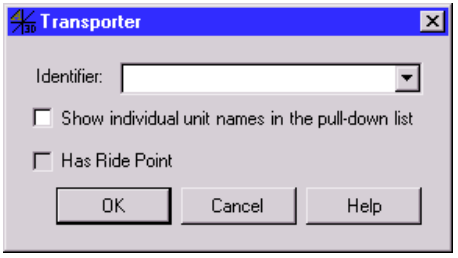
There are two kinds of transporters in Arena: *free-path* and *guided*. Free-path transporters move from station to station on Distances. This movement is not constrained by the presence of other transporters moving in the same Distance Set. Guided transporters move from intersection to intersection on Network Links. Their movement is constrained by the presence of other transporters moving in the same network. (For more information, consult the Arena documentation.)

The Playback data is specific to which type of transporter and movement system is being used. If you're building an animation without benefit of a playback file from a completed model (not recommended), then you need to keep this in mind and not build a network to animate transporters that move in the simulation on a distance set, or vice versa.

The remainder of this section applies to both types of transporters.

Adding a transporter

To add a transporter to the animation, you select **Tools > Add 3D Objects > Transporter** or press the **Transporter** button ( ), which opens the Transporter dialog box, shown below.



If a playback file is open, you can use the Identifier box to choose the transporter from the list. Otherwise, you type in the name of the transporter you want to animate. The association between a transporter and its distance set or network is part of the simulation and does not need to be specified in Arena 3DPlayer.

Once you click **OK** to accept the Transporter Identifier name, the pointer changes to a cross hair. Then click anywhere in the 3D grid to place the default transporter shape. This location gives the shape a place in your layout for Layout Editor purposes only. (Transporters are initialized in the simulation to start at defined stations or intersections when you run the animation.)

You can assign a shape to each state of an animated transporter shape set (which could be a shared or individual unit shape set; see below) using the Shape Manager. Enter this mode by right-clicking on the transporter in the 3D pane and selecting **Choose Library Shape**, or choose **Tools > Shape Manager** and choose the Transporter from the Shape Manager tree pane. To use the tree selection method, the Transporter must be present in a playback file that you have opened.

If you want to view but not assign the shapes associated with different states, you can do that from the Layout Editor by right-clicking on a selected Transporter, clicking **Transporter** on the shortcut menu, and clicking one of the state names.

Once the transporter is visible in the 3D pane, you can bring up the Edit pane (to manipulate its translation, rotation, and scale) by click-selecting or tree-selecting the transporter. Changes to any of these will apply to all of the transporter states. You can double-click in either place to reopen its Transporter dialog box.

SHARED VERSUS INDIVIDUAL UNIT SHAPES

For a given transporter, there can be more than one transporter unit in the simulation moving on a given distance set or network. In these cases, you have a choice between representing each unit with the same (*shared*) set of state shapes, or giving them *individual* shape sets. Normally, your transporter will have shared shapes.

Checking the "Show individual unit names in the pull-down list" check box in the Transporter dialog box allows you to assign an individual shape set to a given unit. (If you are working without a playback file, type the name of the transporter, then add (*n*) to the name where *n* is the unit number you want—making sure not to insert a space before the left parenthesis).

Transporters that have some but not all individual unit shape sets and also a shared shape set defined will use the shared shape for any units that do not have their own shape sets. If there is no shared shape set, then units without an individual shape set will not be animated. For example, an Arena model contains a transporter called Truck with four units. In 3DPlayer, if you define shape sets for *Truck(1)* and *Truck(2)* only, then only those two units will be animated. However, if you also define a shape set for *Truck*, then all four units will be animated. Truck units 3 and 4 will use the shape set defined for *Truck*.

If a playback file is open, the Tree View, if expanded, reflects the shared or unit shapes that have been matched. Note that if all unit shapes have been matched but the shared shape has not, the shared shape will be purple to show that the work is finished, even though the shared shape itself might not have been matched.

Ride points

A *Ride Point* defines the location and orientation, relative to the transporter, of an entity that is riding on the transporter. The entity's (0, 0, 0) point is placed at the transporter's ride point.

The state shape that is shown by default in the Layout Editor for a transporter is its Busy state shape so you can attach your ride point with the most appropriate shape visible (because a transporter is busy when an entity is riding on it).

To attach a ride point, you must first select the transporter so that it turns magenta, then right-click and choose **Ride Point**. With the cross hair, you can place the ride point anywhere near the transporter. Once it is placed, it will move with the transporter if you translate or rotate the transporter in the Layout Editor, and any riding entity will do the same when the transporter moves during an animation. (The ride point itself is invisible in Run mode.)

You can edit the ride point alone by clicking on it in the 3D pane (be sure the transporter is not also selected when you do this). The graphic representing a ride point, like other markers (stations, intersections, midpoints, and arrows), can't be scaled. If you accidentally get your ride point stuck inside the transporter, you can temporarily shrink the transporter and move the ride point out or use **Tools > Layout Settings > 3D Layout Grid Settings** to increase the width spacing of the grid temporarily, which also has the effect of increasing the marker graphics sizes.

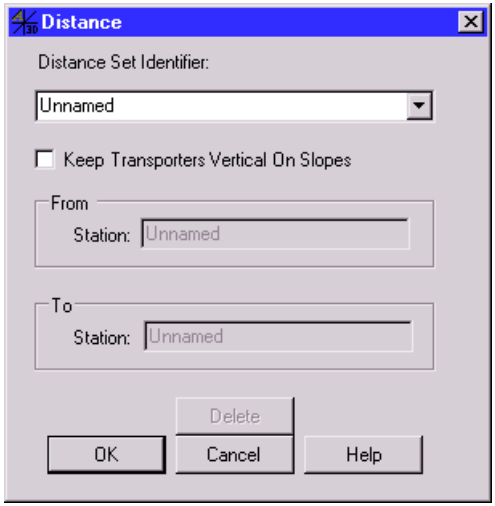
If you have defined separate shape sets for individual transporter units, they each have their own ride-point definition.

Distances (for free-path transporters)

In Arena, free-path transporters move from station to station. In Arena and in Arena 3DPlayer, the animation construct that defines this movement path is called a *Distance*. The collection of distances on which a given transporter can move is called a *Distance Set*. The distance sets and their names are present in the playback file. A free-path transporter is one that does not compete with other transporters in the same distance set for travel paths and destinations.

Distances behave like routes in that they do not have names, and each one connects a pair of stations. Free-path transporters in the simulation move from station to station. In the animation, these transporters travel from station to station on distances.

To draw a distance, you select **Tools > Add 3D Objects > Distance** or press the **Distance** button ( ). The Distance dialog will appear, showing unnamed From and To station names. The Keep Transporters Vertical On Slopes check box allows you to specify that transporter shapes remain vertical when they move along sloping guide paths. This is helpful to represent instances such as a carrier hanging from a chain conveyor.



You identify the distance set to which this distance will belong via the Distance Set Identifier box by typing a name or selecting from among the distance sets, if a playback file is open.

The next step is to click **OK** to begin drawing this distance path. Using the cross-hair pointer, click anywhere in the layout to locate the distance where the click intersects the $Y = 0$ plane. The first click establishes the point at the front of the distance path. You can add points by clicking repeatedly. To finish, you can double-click to add the last point or press the **ESC** key. The display then consists of one white point at the beginning station and a white point at the end station, connected by red lines to one or more black midpoints. (For more detailed information, refer to “Path Drawing and Editing” on page 85.) You can click on an existing station for the start and/or end point; otherwise, a new station will be created.

After the distance has been drawn, you can change the name of the starting and/or ending station. To open the Station dialog box to change the name, you must double-click on the white shape representing the station.

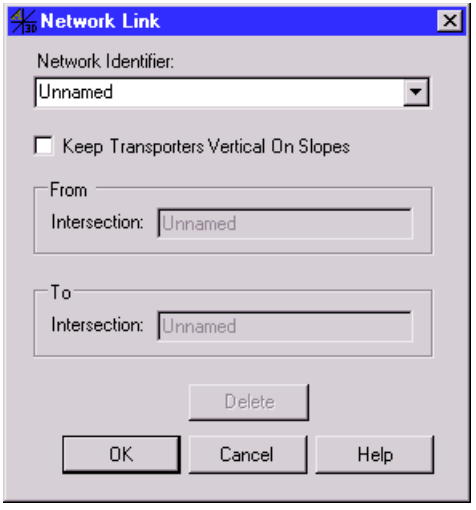
To see the Distance dialog for an existing distance, you simply double-click on the line representing any part of the distance (but not on a station or midpoint). From there, you can change the name of the distance's distance set. You can also delete the entire distance from that dialog box (the stations will remain in case they are needed by other material-handling constructs, and they would need to be deleted separately).

Networks (for guided transporters)

In Arena, guided transporters move from station to station, which are associated with an intersection. In the animation, these transporters travel from intersection to intersection on networks. In Arena and in Arena 3DPlayer, the animation construct that defines this movement path is called a *Network Link*. The collection of network links on which a given transporter can move is called a *Network*. The networks and their names are present in the playback file. A guided transporter is one that competes with other transporters in the same network for travel paths and destinations.

Networks behave like routes in that they do not have names and each one connects a pair of intersections. Guided transporters in the simulation move between stations, which are associated with intersections.

To draw a Network Link, you select **Tools > Add 3D Objects > Network Link** or press the **Network Link** button ( ). The Network Link dialog box will appear, showing unnamed From and To Intersection names. The Keep Transporters Vertical On Slopes check box allows you to specify that transporter shapes remain vertical when they move along sloping guide paths. This is helpful to represent instances such as a carrier hanging from a chain conveyor.



If a playback file is open, the Network Identifier box lists all networks that are in the playback. You may select a name from the list or you may type in the name of the network to which this network link will belong.

Once you have chosen or typed in the name, you then click **OK** to begin drawing this network link. Using the cross-hair pointer, click anywhere in the layout to locate the network link where the click intersects the $Y = 0$ plane. The first click establishes the point at the front of the network link path. You can add points by clicking repeatedly. To finish, you can double-click to add the last point or press the **ESC** key. The display then consists of one yellow point at the beginning intersection and a yellow point at the end intersection, connected by lines to one or more black midpoints. (For more detailed information, refer to "Path Drawing and Editing" on page 85.) You can click on an existing intersection for the start and/or end point; otherwise, a new intersection will be created.

After the network link has been drawn, you can change the name of the starting and/or ending intersection by double-clicking on the yellow shape representing the intersection and changing the name in the Intersection dialog box.

To see the Network Link dialog box for an existing network link, you double-click on the line representing any part of the network link (but not on an intersection or midpoint). From there, you can change the name of the network link's Network Link Set. You can also delete the entire network link from that dialog (the intersections will remain in case they are needed by other material-handling constructs, and they would need to be deleted separately).

Directionality

Normally, your material-handling paths will have a natural direction—particularly for segments and network links. When drawing fresh material-handling guide paths in Arena 3DPlayer, you will typically make the beginning station the entry point and the ending station the exit point for movement along the path. You can review the direction using the dialog for the path, which shows the beginning and ending stations (or intersections).

What happens if your drawing or imported Arena layout has paths that are drawn in reverse? If an entity (or transporter) needs to go from Station A to Station B, but there is only a path from Station B to Station A, Arena 3DPlayer will find that path as an alternate choice and will use it. If both paths exist, the normal path from Station A to Station B will be used.

Stations

You can add *Stations* from outside the path-drawing tools. They will be used in the animation if you later connect them using routes, segments, or distances.

To add a station, you select **Tools > Add 3D Objects > Station** or press the **Station** toolbar button ( ) to open the Station dialog box, shown below.

In the dialog, the Station Identifier box lists those stations that have been defined in the playback (these are also the stations that are listed in red in the Tree View pane, meaning they need to be placed in the layout). You may choose a station name from this list or you may type in the name of the station. Arena 3DPlayer allows multiple animated stations with the same name to be used in different material-handling paths.

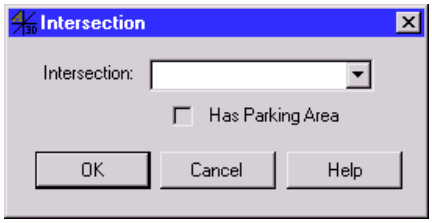
There is an unavailable check box labeled Has Parking Area that is a status indicator telling you whether there is already a parking area associated with this station. Adding or deleting a parking area is a separate function, so you can't change this check box from this dialog box. (Unlike Arena, Arena 3DPlayer does not support "automatic" parking areas because they would frequently end up inside the 3D station marker and would be difficult to manipulate. During animation, Arena 3DPlayer will continue to display a transporter that comes to rest at a path endpoint, even if the station has no parking area.)

Once you click **OK** to accept the station name, the pointer changes to a cross hair. A click anywhere in the layout places a white shape representing the station at the location where your click intersects the $Y = 0$ plane.

Intersections

Intersections are like stations except they are used only with networks. You can add intersections from outside the path-drawing tools. They will be used in the animation if you later connect them using network paths.

To add an intersection, you choose **Tools > Add 3D Objects > Intersection** or press the **Intersection** toolbar button ( ) to open the Intersection dialog, shown below.



The Intersection dialog has a box for entry of the intersection name. If you have a playback file open, you can click the arrow and choose from a list of those intersections that have been defined in the playback. (These are also the intersections that are listed in red in the Tree View pane, meaning they need to be placed in the layout.) Arena 3DPlayer allows multiple animated intersections with the same name to be used in different material-handling paths.

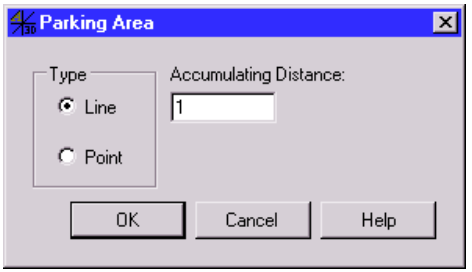
There is an unavailable check box labeled Has Parking Area that is a status indicator telling you whether there is already a parking area associated with this intersection. Adding or deleting a parking area is a separate function, so you can't change this check box from this dialog. (Unlike Arena, Arena 3DPlayer does not support "automatic" parking areas because they would frequently end up inside the 3D intersection marker and would be difficult to manipulate. During animation, Arena 3DPlayer will continue to display a transporter that comes to rest at a path endpoint, even if the intersection has no parking area.)

When you are satisfied with the name of your intersection and click **OK**, the pointer changes to a cross hair. You can click anywhere in the layout, and a yellow pyramid representing the intersection will be placed at the location where your click intersects the $Y = 0$ plane.

Parking areas

Parking Areas display transporters that are stopped at a station or intersection. Each parking area is associated with a specific station or intersection. The parking area determines the location(s) and orientation(s) of the transporters. In Arena 3DPlayer, if there were no parking area, then free-path transporters parked at that station would not be displayed. However guided transporters parked at intersections would be displayed.

To add a parking area, you choose **Tools > Add 3D Objects > Parking Area** or click the **Parking Area** toolbar button ( ) to open the Parking Area dialog box, shown below.



The Parking Area dialog allows you to specify the type of the parking area as Line or Point (see Chapter 7, "Path Drawing and Editing," for more information). If Line is selected, you can also specify the Accumulating Distance, which is the fixed linear distance between the placement of entities or transporters on a given residence-type path. If there is more than one transporter at a Point-type parking area that contains only one point, then the transporters will appear on top of one another.

Once you have chosen the type and clicked **OK**, the pointer changes to a cross hair. Next, you click on the graphic for any station or intersection that does not yet have a parking area. This establishes the association and a white line is shown on the screen that moves with the pointer. Your next click establishes the front endpoint of the parking area. You can add points by clicking repeatedly. When you are finished, you double-click to add the last point or press the **Esc** key. The display then shows a yellow line connecting the front endpoint to the station or intersection. A line-type parking area may consist of one or more points (in addition to the association point).

Path Drawing and Editing

This chapter applies to the four “guide-type” paths (routes, segments, distances, and networks) and the four “residence-type” paths (queues, seize areas, storages, and parking areas), except as noted. The drawing and editing interface is shared among all eight.

Drawing a path

To draw a path, you click on the appropriate toolbar button, navigate through the construct-specific dialog box to get a cross-hair pointer, then click anywhere to place the first point on the “ground level.” You then continue clicking to create a path that is structurally a polyline. When you get to the last point, double-click or else click once, then press **ESC**.

The path dialog has an option to control whether entities moving on an inclined path will “tilt” or not. Sometimes you want entities to remain vertical (such as a person walking, a helicopter flying, or a hanging power-and-free carrier) and other times you want them to follow the incline of the path (a bus going up a hill or a box on a belt conveyor). There is a global option that you can override for a given path from the path dialog. These settings are saved with the layout.

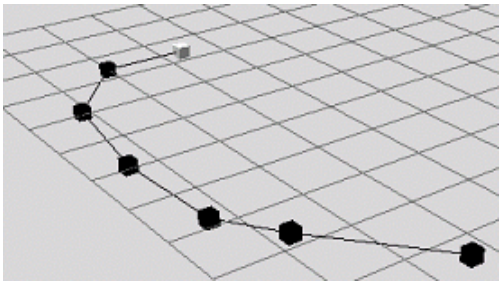
For making multiple guide-type paths in a row, there is a valuable shortcut that allows you to bypass the dialog box for each new path; you right-click instead of double-clicking on the last point of the path. When you do this, the previous path is ended just as though you had double-clicked, but the pointer remains a cross hair and you are able to place the beginning point of a new path of the same type (it will have the same Segment Set, Distance Set, or Network identifier, in the case of those three path types).

Residence-type paths

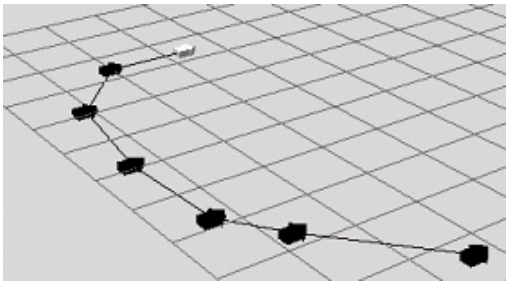
For residence-type paths, the marker for the first point clicked (the front of the line) is white and the last point (if there is more than one) is black. The markers of any midpoints (points where the line can bend) are also black. In the case of parking or seize areas, there must be two or more points, but the front point is the “anchor point” (that attaches to the resource or station, respectively), meaning there can be as few as one point in use. For point-type storages and queues, having only one point is allowed.

If the residence-type path object is line-type, the shape of the marker is a box, and if the path object is point-type, it is an arrow. Both types are shown below. (To be sure the arrows show up in the pictures shown here, the markers were magnified for the residence

paths by increasing the width grid spacing using **Tools > Layout Settings > 3D Layout Grid Settings**.)



Line-type residence path



Point-type residence path

LINE-TYPE RESIDENCE PATHS

As the pictures above show, residence paths come in two types: Line and Point. Line-type paths use distance-based accumulation on a continuous polyline to separate the entities. Unlike Arena, which uses the picture geometry for spacing on a line-type path, Arena 3DPlayer uses a fixed linear distance that you specify for a given path. You specify this distance in the dialog box using the same distance units you are using elsewhere in the layout.

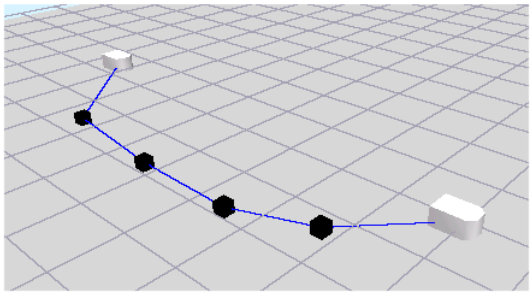
Entities on a line-type residence path always face ahead (parallel to the line). Note that if an entity happens to land exactly on a point in a line-type residence path, the orientation is taken from the segment behind. If there is no segment behind (that is, it's the last point in the path), then the orientation is taken from the segment ahead.

POINT-TYPE RESIDENCE PATHS

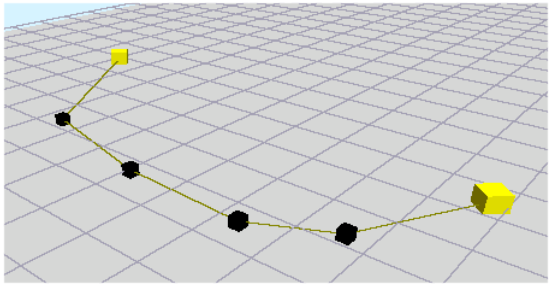
While on point-type paths, entities wait at each junction (midpoint) in the polyline, never in-between. Arena 3DPlayer offers complete control of the orientation of entities that come to rest on a point in a point-type residence path. Each point has its own orientation. You can edit the orientations as a group (using graphical multiple selection) or individually.

Guide-type paths

For guide-type paths, every path must have at least two endpoints. The endpoints are always stations (for routes, segments, and distances) or intersections (for network links). Stations are shown as white shapes, and intersections as yellow boxes. In either case, these serve as the ending markers, but they are not "part of" the path (they could be shared with other paths). The markers for the midpoints, if any, are black, as they are for residence paths. Below are pictures illustrating guide paths.



Guide path (Route, Segment, or Distance)



Guide path (Network Link)

Editing a path

There are three main things you can do to a whole path: rename it, delete it, or move it.

To rename any type of path, double-click on one of its lines (not on a point) to bring up the dialog box for that path type. The lines and points will all turn magenta (except endpoints of a guide-type path). You can type in a new name here.

To delete a residence-type path, select all the points by single-clicking on one of the segments, and delete them using the **Delete** key or **Edit > Delete** or the **Delete** command in the shortcut menu (accessed by a right-click). The path will be gone.

The easiest way to delete a guide-type path is to double-click on one of its lines to bring up the dialog box and click **Delete**. The path will be gone, but the ending stations or intersections will remain (because they might be used for other paths). If instead of using the dialog, you try deleting all the points in a guide-type path, you will be left with a single-line path connecting the endpoints. You can single-click that line to turn it magenta (this "no-midpoint path selection" is a special case) and press the **Delete** key to complete the deletion. Or, if you wanted to delete the endpoints anyway, you could have added them to the selected point set using **CTRL+Click** prior to the first delete, and the whole path would be gone.

To move a whole path, click once on any line segment in the path (not on a midpoint or endpoint). This selects all points in the path as a multiple selection. Then move the multiple selection by dragging any one of the selected (magenta) points or by changing the XYZ values on the Transformation tab. For a guide-type path, the ending stations or intersections will not be included in the selection (just like Arena), but you can include them by using **CTRL+Click**.

Editing path points

In addition to the whole-path operations described above, you can manipulate the individual points on a path.

If the endpoints and midpoints are too small to click easily, modify your width grid spacing to a larger value. Arena 3DPlayer sets the graphic sizes for markers automatically relative to the width grid spacing.

You can select any path point by clicking on the point, and the point will turn magenta just like any object. You can delete, translate, rotate, or scale the point. However, rotation has no meaning except for the points in a point-type residence path (which are shown as arrows), and the numeric scale never has any meaning for a marker-type object (marker-type objects relating to paths are stations, intersections, and the boxes or arrows used to represent path midpoints and also residence-type path endpoints).

When any midpoint, or the beginning point of a residence-type path, is selected, you can access the Insert Points command from the shortcut menu. Your pointer becomes a cross hair, and you may add one or more points after the selected point with each click. A double-click ends the insertion, or simply press the **ESC** key.

You can also multi-select points for deletion or translation. Basic multi-selection methods were described in the section on "3D multiple selecting and editing" on page 39, using the **CTRL** key for adding/subtracting single points or the **ALT** key to select points within a region. The other two group-selection ways that only apply to paths are:

- Clicking on one point then holding down the **SHIFT** key and clicking on another point on the same path, which selects those two points and all points on that path that are between them.
- Clicking once anywhere on one of the line segments, as described earlier in "Editing a path" on page 88. This is a shortcut for "multi-select all points in this path." It will not select the endpoints for a guide-type path.

All of the selection methods described here select points—not the path itself.

8 Importing Arena Animation Information

General import considerations

Special features exist in Arena 3DPlayer to make it easy to import existing animations from Arena. There are two reasons for this:

- You might want to convert existing Arena animations to 3D.
- Although the Arena 3DPlayer environment is similar to Arena, when you are new to Arena 3DPlayer, you might choose to start out a new animation using the familiar 2D Arena layout creation environment.

Of course, you may prefer to develop your 3D animations from scratch in Arena 3DPlayer.

The distinction between what is “simulation information” and what is “animation information” is the same in Arena 3DPlayer as in Arena. For example, the speed of a conveyor is “simulation information,” and therefore is determined through Arena’s simulation modeling interface, not through the drawing interface in either Arena or Arena 3DPlayer.

As another example, the starting time and ending time of an entity routing from one station to another is defined in the simulation, where the path orientation of the route is defined in the animation.

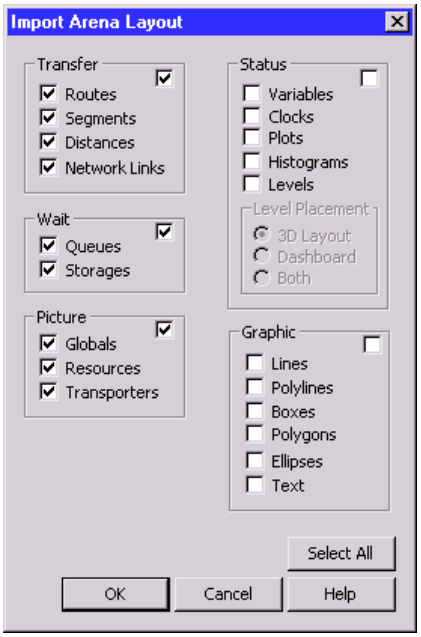
Comparing Arena 3DPlayer with Arena’s “two-step” animation mode, the basic building blocks are very similar in Arena and Arena 3DPlayer.

If you are importing an existing 2D Arena animation as the basis for the 3D animation, select **File > Import Arena Layout**. Imported animation information comes in via Component Object Model (COM), a live link to an open Arena model. Arena must be running and the model must be open before you initiate the import.

The default settings of the Import Arena Layout dialog box, shown below, reveal that Arena Transfer, Wait, and Picture information are automatically selected for import. Manual selection is needed for Status items, such as Variables, Clocks, Plots, Histograms, and Levels, and for Graphic components of the animation.

When Levels is selected, additional options are displayed that determine the placement of the levels in either the 3D Layout, Dashboard, or both. If the 3D Layout option is selected, all Rectangle levels from Arena will appear in the 3D layout as cylindrical-shaped tanks. If the Dashboard is selected, all Rectangle levels from Arena will appear as Rectangle levels in the Dashboard. If the Both option is selected, all Rectangle levels from Arena

will appear in both the Dashboard and 3D layout. Circle and Dial levels will appear in the Dashboard only, and Flow and Distributed Fills will appear in the 3D layout only.



Arena 3DPlayer always imports an Arena layout into a new Arena 3DPlayer layout. You cannot add onto an existing layout by importing from Arena. (If this need arises, you can use Arena 3DPlayer to cut from another layout and paste material into your newly imported layout.) If you plan to import an Arena model, it is advisable to do so before any CAD (DXF) importing or other Arena 3DPlayer work is done.

Arena 3DPlayer attempts to match the Picture ID for any resource, transporter, or global state from Arena to an existing shape in Arena 3DPlayer's shape library. Arena 3DPlayer first attempts to find an exact match for a given Picture ID in its shape library. If no match is found, Arena 3DPlayer tries to match the shape category and group, and if successful, it assigns a shape to the first instance in this group. For example, if the imported Picture ID is *Shape.Ball.Pink* and there is no match in Arena 3DPlayer's library but there are one or more shapes whose names begin with *Shape.Ball*, the resulting picture assignment will be the first of those shapes. To support this feature, 3DPlayer now loads the shape libraries when importing an Arena model if they have not already been loaded during the current session (normally a task performed on the first invocation of the Shape Manager).

The Arena Import progress dialog box contains separate progress bars for each type of item imported from Arena. This enables you to monitor the progress visually to determine the stage of the entire import operation.

An import from Arena brings in nearly everything in the animation, plus static Arena shape graphics, including:

- Geometry of linear constructs (path-type constructs—routes, segment sets, distance sets, and networks—and also queues, seize areas, storages, and parking areas)
- Geometry of static shape graphics (lines, polygons, etc.)
- Placement of resources and globals (and initial placement of transporters)
- Entity picture names
- Other constructs that are naturally 2D (text, clock, variable, etc.)—placed in the Dashboard
- Levels (which you can direct to the 3D layout as tanks, to the Dashboard as Arena-like levels, or both)

Dashboard import considerations

In general, you may find the import of Arena information into the Dashboard to be of limited use. That is one reason why, by default, the check boxes in the Import Arena Layout dialog are set up to ignore some information. The reasons for this are:

- Shape graphics other than text (e.g., boxes and polygons) are sent to the 3D layout, not the Dashboard.
- Font sizes are not imported if you are using a version of Arena prior to Arena 8.0.
- Tables and descriptions are often intermingled with the animation in Arena; in 3DPlayer, a different layout approach is usually desirable.
- You may find the imported text difficult to work with because it comes from a mixed animation-and-text environment. You may prefer to start your Dashboard display from scratch.
- Remember that when Levels is selected and the Dashboard is specified to receive the items, all Rectangle levels from Arena will appear as Rectangle levels in the Dashboard. If the Both option is selected, all Rectangle levels from Arena will appear in both the Dashboard and 3D layout. Circle and Dial levels will appear in the Dashboard only.

Tutorial 7: Importing and running an Arena animation

In this tutorial, you'll work with importing an animation from an existing Arena model file.

1. Start Arena and open the **tutorial7.doe** model (located in the Tutorials sub-folder of the Arena program folder). This is a simpler version (using fewer Entity Pictures) of the model used to create the Banking Transactions animation in the Examples folder.
2. Start Arena 3DPlayer and select **File > Import Arena Layout** to open the Import Arena Layout dialog box.

   With the array of check boxes available, you may select the individual animation components you wish to import (e.g., if you don't want to import static shape graphics), or you may click **Select All**. For this tutorial, go ahead and choose **Select All**, then click **OK**.

   After a brief delay, your imported layout will appear. You should recognize the basic layout of the bank as represented by polygons and boxes (which are represented by very thin 3D static shapes a very short height above the floor). All the routes, queues, and resources (with seize areas) will be present.
3. Save your newly imported layout in the Tutorials folder as **tutorial7.a3d**.
4. If you haven't already done so, from within Arena 3DPlayer, open the **tutorial7.pbf** playback file that is in the Tutorials folder by using **File > Open Playback**. (You could also regenerate this file from Arena, but the time units in the supplied Arena model might need some adjustment.)

Working with resources

Your layout and playback files should both be open, so now you can use the Shape Manager to assign reasonable person shapes to the two walk-up teller and the one drive-through teller resources, and machine shapes to the automated teller machine (ATM).

You could choose the resources in the Shape Manager by the method recommended in Tutorial 5, but here's a different option.

1. Start in the Layout Editor. Select a resource by clicking on it once. It will turn magenta color indicating it is an editable shape. Then right-click and select **Choose Library Shape** from the shortcut menu; the resource will already be in focus.
2. Because resources have multiple states (built-in and possibly user-defined), you should attach shapes that will let you differentiate among them. In this case, you'll only define the busy and idle states, so choose the blue-shirted "man with counter" shape to represent both states. Once the shapes have been assigned and you've

returned to the Layout Editor, re-select the resource and change the shirt color of the busy resource to green via the Color tab of the Edit pane in order to differentiate between the two states.

3. Repeat Step 2 for the other resources.
4. If you have a resource that is facing the wrong direction, you can change it. From the Layout Editor, click on the resource (if needed) so that it turns magenta. In the Edit pane below, locate the +90 and -90 buttons at the top right. Experiment with these buttons to turn your resource in different directions.

Working with seize areas

Once you're satisfied with the orientation, shape, and location of your resources, it's time to modify the seize area for each resource. In each case, you want a one-point point-type seize area, and this is what was imported from Arena.

1. Because this is a point-type seize area, you will see an arrow at the seize point. Go ahead and rotate the arrow (click on the arrow only, making sure the resource is not also selected) and rotate the arrow so that it points in the correct direction.
2. You want your bank customers to face the tellers, so rotate the arrows as needed to point at the tellers. The seize area at the drive-through is a special case. You want the arrow to point forward, because a vehicle is served from the side.

Working with queues

The customer queue inside the bank lobby is a point-type queue in Arena. If you leave it point type in Arena 3DPlayer, you'll have to rotate each point to a suitable orientation, so make the following change:

1. Even though you could group-select and rotate them in batches, it's easier to convert the whole queue to an Arena 3DPlayer line-type queue.
2. Specify an Accumulation Distance of **2** or **3** units (feet, in this case). (To bring up the Queue properties dialog box, double-click on any segment of the queue.)

Working with entity shapes

Make the following changes to the entity shapes:

1. Assign a minivan shape to the "cars" entity picture name.
2. Choose any other two People shapes for the ACustomer and TCustomer entity picture names (ATM and Teller customers, respectively).

Analyzing the animation

At this point, you're ready to try out the animation. When it runs, you'll find that there are some further things you may want to clean up.

1. As it runs, watch for things like the location and nature of the queue points at the ATM and Window Teller, and the route leading to the Window Teller queue (which has an undesirable "hook" at its far end in the base model). You don't want the vehicles to overlap or to flip around due to the hook.
2. End the animation run to improve the positioning and flow of the items mentioned above.
3. Re-run the animation by clicking the **Go** button, and view your improved animation!

What else can I do with this animation?

There are many other things you could do with this layout. You might notice that the overlapping boxes and polygons in the Arena model are imported as thin wafers at the same *Y* (height) location in Arena 3DPlayer. This can lead to a distracting interference effect when the 3D rendering engine tries to differentiate points on these two (or more) coplanar surfaces. But you can click-select one of the shapes and move it up or down slightly to improve this. If you move the ATM floor down, you might even uncover the 2D static people from the Arena model.

The imported layout is near but not at the coordinate origin (the 0, 0, 0 point, which by default is where the grid and built-in "floor" are centered). You might want to shift the gray floor horizontally to be under your layout, or move the layout horizontally to be above the floor. You can do either (using **Tools > Layout Settings > 3D Layout Grid Settings** to move the grid and floor, or by region-selecting the entire layout and relocating it, respectively). If you decide to move the layout, you'll have to do it in two passes because you can't group-select static shapes and other objects simultaneously.

You might want to fine-tune the routes and queues by moving points around individually or as a group. You can add points to a route as described in Chapter 7, "Path Drawing and Editing."

You might also use this as a test bed to try your hand at drawing a wall system. If you build walls and other shapes into a new static system, you can eventually try making invisible the static system that was created during the Arena import. This may seem easier than going through and deleting the various "extra" shapes that were imported with the Arena layout.

Enhancing an Animation

It's possible to make a 3D animation that is beautiful or one that is boring. Often this is a reflection of the time that is available, but Arena 3DPlayer has many features to help you make more appealing animations in less time. Some of the possibilities are covered in this chapter.

To start the discussion, it's helpful to think of the animated view in three components: live, static, and statistical. The statistical component is handled in Arena 3DPlayer using the Dashboard (see Chapter 5). The live and static components appear in the 3D pane, which is the focus of this section.

Live components

The live components in 3D are the animated representations of entities, resources, globals, and transporters. Each of these has the ability to move or to change shape.

It is possible to further subdivide the discussion into resources and globals (which can change shape but never move) and entities and transporters (which can move).

For resources and globals, it is recommended that you use shapes that are realistic. Arena 3DPlayer comes with over 150 predefined shapes. If you can't find a predefined shape that matches your need, you have two choices:

1. Find and convert an existing 3D picture from an equipment vendor or other source such as the Internet (be sensitive to permission issues if your animation will be publicly displayed or provided to others).
2. Draw it yourself using a drawing program or Arena 3DPlayer's Shape Editor (or have someone draw it for you).

For bringing in new shapes that you find or create, you need to find, save, or convert the shape in such a way that it ends up in a compatible VRML format, and then use Arena 3DPlayer's VRML-to-user-library tool in the Shape Manager. This process is described in Chapter 4.

If you wish to use colors to represent resource states in shapes that you draw, consider following certain conventions for this as is done for the examples: blue for idle, green for busy, red for failed, and purple or cyan for other states you may choose to define, such as

blocked or setup. Just as in Arena, you need to locate or create a separate shape for each state.

Whenever you save your *.a3d* file, all shape definitions that you have assigned are saved in that file. This means that your model is never dependent on the current contents of the standard library or user library files, and you do not need to include shape library files when you hand off an Arena 3DPlayer animation to someone else. (There is a tradeoff to this approach. Because each assignment places a separate copy of the shape in the data structure and, when you save, in the *.a3d* file, this also means if you use the same library shape over and over in a model, it is not possible to change the library definition to have the changes propagate through.)

Most of the same considerations apply for entities and transporters. Remember to take full advantage of the control you have over the position and orientation of entity placement points such as ride points, seize areas, point-type queues, and storages, etc. For best animation performance, avoid using highly detailed 3D shapes for entity picture that will be displayed hundreds of times simultaneously. (Shapes with curved surfaces should be considered to be more complicated than boxy-looking shapes.)

In addition, you will need to be sensitive about how you use queues and storages. Arena does not combine its Queue/Storage constructs with any movement-on-path rules, so you do not have the option to have something "move smoothly through a queue" unless you replace your queue with a material-handling construct.

Static components

One of the best ways to add depth and interest to a 3D animation is through the use of static components. You don't want your animation to appear too "busy," but at the same time, you can enhance the realism this way with not too much effort.

Your choice of static components depends on the nature of your animation. For retail settings such as a bank or a restaurant, you may want to add furniture and equipment and even decorations such as signage or plants. For a manufacturing environment or warehouse, a CAD-like approach may be appropriate. For medical facilities, where the concept of room-as-resource is important, you may want to focus on the floor plan with walls to depict the rooms and corridors.

As with entities, you should try to avoid creating dense populations of highly detailed and/or curvy static shapes. Depending on the display hardware, showing hundreds of trees with spherical tops and cylindrical trunks may slow down the animation noticeably. (The palm tree in the standard library is an example of a shape that has been engineered for reduced "triangle count.")

10 Creating AVI Movie Files

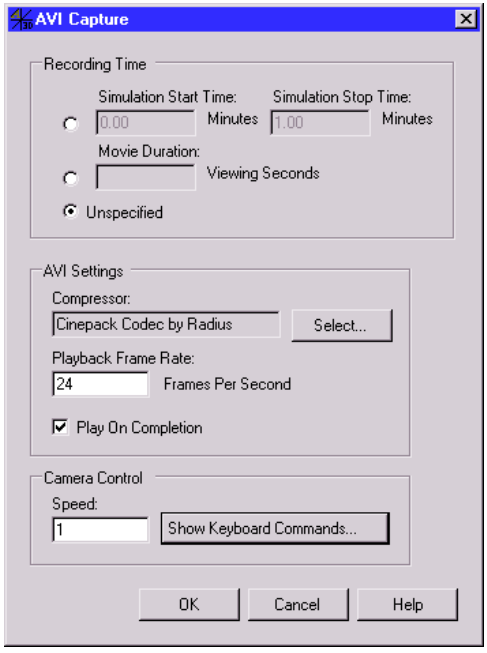
Arena 3DPlayer has a movie-making feature that you can use to create movies of your animations. The supported format is *.avi*. All versions of Microsoft® Windows® include the Windows Media Player that can play *.avi* files. Other players are available as well.

About AVI capture

The interface to the AVI Capture feature is designed to be simple and intuitive. At any time while running an animation, you can start recording a movie of the animation. The movie can be as long or as short as you like. Everything displayed in the current 3D pane will be recorded.

Capturing an animation

The control settings for the movie-making tool are contained on the AVI Capture dialog box, which is accessed by selecting **Tools > AVI Capture** from the main menu or by pressing the **AVI Capture** toolbar button ( ). These options are available when a playback file is open.



The AVI Capture dialog box provides Recording Time, AVI Settings, and Camera Control features. The Recording Time feature allows you to determine the duration of the AVI movie. Clicking the top option button allows you to set the start and end simulation time of the AVI capture. Clicking the second option button allows you to set the movie duration in seconds, and the third option button allows you to record for an unspecified time (or until the animation completes). The AVI Settings feature allows you to choose the codec used for recording and playback and the number of frames per second for movie playback. The Play-on-Completion option determines whether the movie will automatically begin playing in your default media player upon completion of recording. The Camera Control feature allows you to set the speed at which the camera moves when manipulating the viewpoint with the keyboard. Selecting the **Show Keyboard Commands** button invokes a dialog providing the keyboard commands for manipulating the viewpoint.

The AVI capture begins once you click the **OK** button. If a layout and playback file are loaded but Arena 3DPlayer is not currently running an animation, Arena 3DPlayer will begin running the animation and start recording an AVI.

The AVI Capture tool provides the flexibility to start and stop capture at any point in the animation along with having the flexibility to create a movie of a desired duration. The AVI Capture tool continuously computes the duration of the movie by dividing the number of frames captured by the playback frame rate and displays the movie duration in flashing red text in Arena 3DPlayer's status bar. To create a movie of a desired duration, simply click **OK** to close the AVI Capture dialog and begin recording and then click the **Stop AVI Capture** toolbar button or select **Tools > Stop AVI Capture** to end recording once the movie time has reached the desired duration.

Once a capture stops, the Save As dialog appears, prompting you to select a file name and location where you wish to save the AVI file. If the AVI Capture setting "Play on completion" is set, the AVI movie will begin playing in your default media player once saving is complete.

Once you have mastered the art of making movies from a single camera viewpoint, you can consider ways to make the movie more interesting by varying the viewpoint.

Varying the viewpoint

The viewpoint, also called the camera, is the location from which you are viewing the geometry. The entire set of standard 3D navigation controls work while you are recording a movie. These include left-dragging to rotate and tilt, right-dragging to zoom, and middle-click to pan by jumping to a new center point. However, it may be awkward to use the mouse during a recording session (primarily because recording may be taking place at a speed different from the playback speed), so keyboard equivalents are available that move the viewpoint in increments.

Because the keyboard equivalents are incremental, it may be difficult to achieve smoothness while manually operating these controls during a recording session, so there are additional options that cause the viewpoint to move continuously. Note that the rotation and tilt commands are designed to feel that you are moving the geometry (that is, behind the scenes, the camera itself is moving in the opposite direction, so a rotate left actually moves the camera to the right). The keyboard commands are listed in Appendix C, “Keyboard Controls for Camera Movement” on page 119.

You can make the continuous movement faster by repeating the keyboard command or slower by using the “opposite” keyboard command. For example, if you press **CTRL+Right Arrow** followed by **CTRL+Left Arrow**, the viewpoint will move and then stop.

Any use of a different type of continuous camera control, or any stepwise camera control, will cancel the current continuous movement. Another way to stop the continuous movement is to press the **ESC** key.

You can view a help dialog box for these commands by selecting the **Show Keyboard Commands** button on the AVI Capture dialog box. This help dialog is modeless so it can remain visible while a recording is in process.

The default speeds of the continuous camera controls are tuned to provide smooth movement in the playback of an AVI movie. The default speeds may not be “right” for you in all cases. You can control the speed of the continuous navigation controls from the Camera Control feature of the AVI Capture dialog box. The speed factor setting determines the factor by which the geometry or viewpoint moves.

Recording quality

Three things determine the quality of a digital stream: color depth, pixel density, and frame rate. The higher the level, the better your movie will look. However, the associated “cost” of higher quality is represented by larger file size and possible degradation of playback performance (which in turn depends on the computer speed, playback window size, and software being used to play back the movie).

The color depth of your AVI will be 32 bits per pixel.

The pixel density of your AVI will be the exact dimensions of the 3D pane. If you want to change those dimensions, you can do that by adjusting the main window size and/or one or both splitter bars.

The frame rate of your AVI may be anything you choose. Frame rates below about 15Hz will flicker noticeably but may be fine for a simulation animation. Frame rates above about 50Hz generally will look very smooth. The effect you see will depend on the colors and the nature of the movement you are showing, so there is no single “best” frame rate.

The AVI Settings box of the AVI Capture dialog box allows you to choose the video codec to use for data compression, the playback frame rate, and whether you would like the movie to begin playing in your default media player upon completion of the capture.

You can choose the video codec from the list of installed codecs that support the color depth of the AVI by clicking the **Select** button on the AVI Capture dialog box. The default setting is the Cinepak® Codec by Radius, Inc. Of all of the codecs that come installed with Windows, we recommend using either the Cinepak Codec or Microsoft® Video 1 codec. Microsoft Video 1 provides a much faster capture rate, but produces a much larger movie file than the Cinepak Codec.

The codecs that come with Windows represent video compression technology from the early 1990s. Many new codecs that support DVD recording and playback have been introduced that yield improved results in both video quality and data compression. If you install another codec, it will show up in the codec list mentioned above, and you can try using it. Be sure to view any AVI files created using newer codecs to be sure they play back. Be aware that they will not play back on another computer that doesn't have the codec you installed.

Arena 3DPlayer is designed to produce your movie at the exact playback frame rate and playback speed ratio you specify, regardless of your computer's performance capability. Sometimes your computer can't keep up with this process in real time. Although this will make the movie-making process take longer than you expect, the playback results will be the same regardless of the speed of the movie-making computer.

Additional Tips

Locating an object

Arena 3DPlayer provides a unique “highlight” feature that helps you locate an object in a complex layout. If you click once on an object in the Tree View pane, it will merely be selected. However, if you double-click the object in the Tree View, the actual display of the object in the 3D pane “signals” by turning magenta and flashing a temporary size change. For objects that can be selected, double-clicking will also leave the object selected.

Forward motion

If you are defining a shape outside of Arena 3DPlayer, you should set its hot point and direction of motion. The hot point is the 0, 0, 0 coordinate point in the shape’s coordinate space. The direction of motion is along the positive *Z* (depth) axis.

If you encounter a shape that is defined somewhere else with an inappropriate hot point and/or direction of motion, you can try to compensate for that by editing the entity picture (after assigning the picture, make visible the entity you want to edit from the list in the Tree View pane, and edit as you would any 3D shape).

Because positive *Z* points away from you in the Arena 3DPlayer coordinate space “front view,” you may want to use other views when editing the graphic for an entity picture.

Naming conventions

Naming conventions in Arena 3DPlayer correspond with names in Arena. Names must follow the same naming conventions as in Arena. The following restrictions apply: Symbol names can be alphanumeric strings containing letters, numbers, embedded spaces, and these special characters: @, _, %, ?, #, and <period>. Symbol names may not contain any other punctuation characters such as <comma> or <colon> and they must not be reserved words such as COUNT.

Unlike Arena, names in Arena 3DPlayer are case-sensitive. So if you have a resource named “Welder” and add one named “WELDER,” these are considered separate resources in Arena 3DPlayer. Names of library shapes are also case-sensitive.

There are various categories of animation items whose names behave as follows:

Entity Picture, Resource, Queue, Storage	Duplicate names are not allowed.
Globals and Route	Not named.
SegmentSet, DistanceSet, Network	Individual sections (Segment, Distance, or Link, respectively) are not named but are assigned membership in the named set; only one instance of a given ordered station pair is allowed per named set, but there can be multiple instances overall.
Stations, Intersections	Multiple instances with the same name are fully supported, as in Arena; there are no sequentially assigned default names because unnamed instances are taken to be multiple instances of the name "Unnamed."

Unusual rotation axes

Sometimes you might like to rotate an object by 90 degrees and also give it a 90-degree twist. An example might be to take a standard one-point seize area that happens to point north and re-orient it so that you can make a "standing person" library shape lie down (for example, on an X-ray table). If the head of the person will point south, you have no problem—just rotate the point by 90 degrees backward around the appropriate axis so that the point is up and the surface that had been the top of the arrow shape faces south. But if the head must point east, you have a problem. Arena 3DPlayer supports one rotation around one axis only.

The solution to this particular problem is to give each axis in the 3D Edit pane for this object a value of +/- 1, and use a 120-degree rotation. You may have to experiment with +1 and -1 values for each axis, and/or 120- and 240-degree rotations, to get exactly what you want.

Status expressions

Some animation objects, when you define them, call for you to specify a *Status Expression*. These include globals, variables, and levels. All of these objects' dialogs have a drop-down list where you can see all status expressions that appear in the playback file header. The expressions themselves, not their titles (if different), are displayed in the Arena 3DPlayer drop-down lists.

Generally, you will want to have a playback file open before defining these objects. The expression must exist in the playback file in order to be animated. If the playback file is open while you are building your layout, you have a choice of all expressions appearing in

the playback file header. If the playback file is not open, you would have to type the expression exactly and hope it appears in the playback file.

If the expression you want is a valid status expression but it is not in the playback file header (i.e., does not appear in the drop-down lists), go back to Arena and either be sure the expression is used in the Arena animation or make it a user-defined expression via **Run > Setup > Run Control > Status Expressions**. Then regenerate the playback file.

Distance units

Arena 3DPlayer is unitless with respect to distance, so your distance units can be whatever you want—inches, meters, furlongs, etc.

However, if you use the built-in standard library shapes without scaling them, you will see that they are dimensioned in feet. For example the "People" shapes in the standard library are about 6 units (feet) tall. If your layout is to-scale and is drawn in meters, for example, you'll need to scale the standard shapes that you use by about 0.3 (to make a 6-unit man into a 1.8-unit man). If your layout is in inches, then you'll need to scale the entities by 12 (to make a 6-unit man into a 72-unit man).

You will also need to consider this effect if you are building wall systems using the built-in wall shapes in the Building category against a floor plan that is in inches or meters. The expandable wall section is 1.0 unit long and 8.0 units high, so if you are matching up with a floorplan that is in meters, you would want to scale the height and thickness by 0.3. You would use the number of meters directly as the *X* coordinate. Once a single wall segment is modified in this way, you'll find it easy to use that section as the "master" and simply copy it and paste enough segments to create the rest of the wall sections, rotating 90 degrees whenever needed.

Many layouts, including some imported from Arena, are not drawn to a uniform scale. In those instances, you will need to scale those shapes to conform with the standard you define in your new animation.

Lighting

Arena 3DPlayer includes both bright directional and dim ambient lighting so that you can see your geometry from any angle. The intensities are pre-set to provide a pleasing balance. The directional light shines from the "front" (actually up high in one of the front corners) and from an infinite distance (but it does not cast shadows).

If you need to look at your drawing "from behind" or "from below," you might find the shapes hard to distinguish because only ambient light reaches these surfaces. You can address this on a temporary basis by selecting **View > Attach Light Source To Camera**. Now the directional light source shifts so that it is always pointing "from you" toward the

screen. Different areas of the geometry will "light up" as you rotate the geometry. Note that this setting is not saved with the layout.

Using this mode temporarily may help in certain situations where the display device can't separate dark or dimly lit parts of the animation (for example, when giving presentations with some data projectors).

Shape Drawing: Beyond the Basics

Tutorial 8: Extrusions

Extrusions in Arena 3DPlayer are useful when you want to draw a simple polygon with a 3D look. This could be anything from a triangle to a map. For example, when you import an Arena drawing that contains polygons, they are converted to 3DPlayer extrusions for a 3D look.

In this exercise, we will build a stop sign similar to the regulatory signs used at street intersections in the United States. The typical stop sign is a regular red octagon with a thin white border and the letters "STOP" printed in white across the center of the octagon.

The stop sign will be drawn in the Shape Editor using the Extrusion tool. To make this exercise simple to follow, we have done a little math for you and know that if the sides are 10 units each and the octagon has a total width of 24 units, a nearly perfect octagon is quite easy to draw.

An easy way to use the Shape Editor to draw a new shape is to start with some existing shape. For this exercise, we will use a Static Shape.

1. Press the **Static Shape** button ( ) from the toolbar to place a static shape in the layout (it will be the default "potted plant" shape). Select the static shape and right-click to select **Edit Current Shape**.

   Now you are in the Shape Editor—recognizable by the shape creation buttons on the secondary toolbar, the four tabs in the Edit pane, and an isolated, centered representation of the shape you are editing.

2. Because we want to draw a stop sign shape, we don't need the potted plant geometry so we'll remove it from the shape tree. Click on the "+" sign next to the StaticElement in the tree, then click on Nature.Plant and press the **Delete** key.

   Once the plant is gone, the shape contains nothing but an empty "root-level" group. This group cannot be deleted. There is one multicolored triad representing the transformation of the root-level group.

   In the tree view, the root-level group (StaticElement) is multi-colored, indicating that it is the "active" group and anything you add to the shape will be in that group.

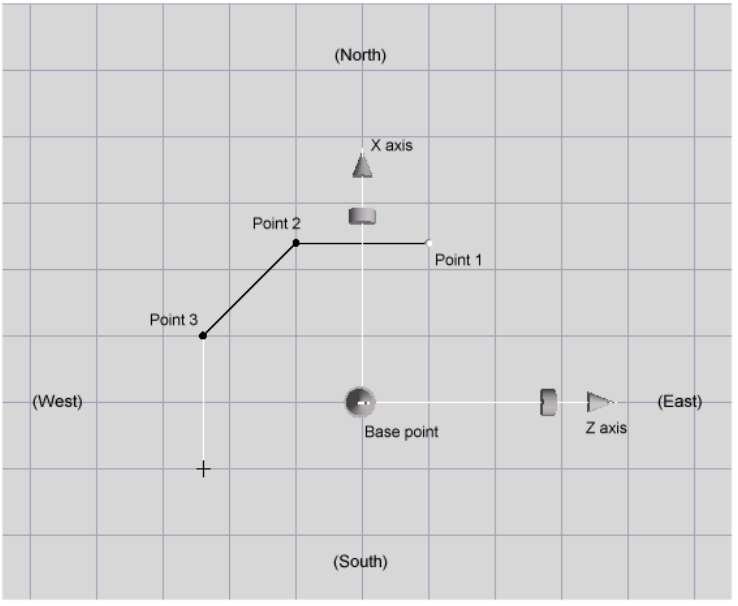
3. In order to draw the octagon shape precisely, turn on the snap-to-grid feature by pressing the **Snap to Grid** ( ) button in the main toolbar. By default, the gridlines are five units apart and the snap increment is one unit.

Because you are drawing a shape with equal-length sides, you might find it helpful to select **View > Normal View > Top** so you are viewing the drawing surface from the top rather than from a perspective view.

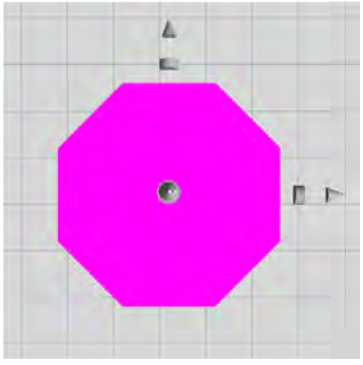
It is also recommended that you zoom in to draw at a close range so that counting the snap increments is easier.

4. Press the **Extrusion** ( ) button in the Shape Editor toolbar. The pointer changes to the "+" cross hair insertion pointer so that you can place the base point of the extrusion. The base point or hot point is not part of the geometry—it only determines where the extrusion is if you move it. The triad for the extrusion will indicate the location of the base point after it is placed, and we will draw our stop sign so that the base point is at the center of the shape.

5. We'll use the base point as the center and draw our points using compass terminology to aid us in creating the octagon shape. The drawing illustrates the orientation and first steps in creating this shape.



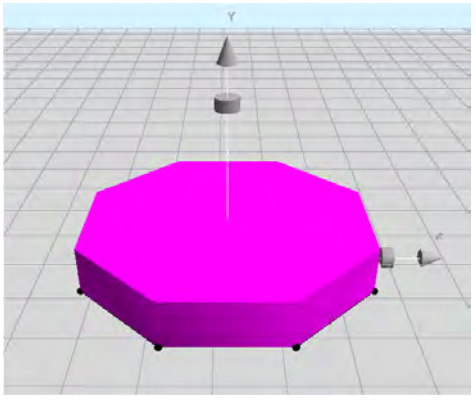
- Click at the intersection of any gridline to establish the base point. A triad will appear at the intersection where you clicked. Now that you have placed the base point (you still have the + pointer), it is time to draw the polyline that represents the border of the octagon.
- Measuring from the base point, we need to move 12 increments north from the center (along the *Z* axis) and 5 increments east. Because we have not yet clicked to place the first point, we won't be able to "count" the snap increments for this first point. Since each unit represents 5 snap increments, simply move the pointer along the *X* axis for two full grid units, plus 2/5 of the distance to the next grid unit. Then move east 5 increments, which is equal to 1 full grid unit. Click. (If you have trouble gauging the exact distance, you can correct the position of any point once the polygon is drawn.)
- Now move due west horizontally for two units (10 spaces). Notice that you can now count the snap increments. Click.
- Without clicking, hover the + pointer at the base point so that we can use the center position as a reference for counting the next segment. From the center, count 12 increments due west (along the *X* axis) and 5 units north. Click.
- From Point 3, count two units (10 spaces) due south. Click.
- Without clicking, hover the + pointer at the base point and count 12 spaces due south (along the *Z* axis) and 5 spaces west. Click.
- Now move horizontally due east for two units (10 spaces). Click.
- Again, return to the center without clicking so that you can count 12 spaces due east along the *X* axis and 5 spaces south. Click.
- Move two units (10 spaces) due north along the vertical line. Click.
- Click at Point 1 (the white dot) to add the last line segment and close the polyline, or simply double-click.



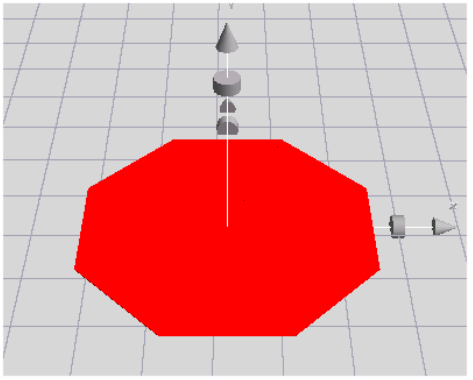
Did you make any mistakes? If your line segments are not of equal length, you can edit the individual corner points so the segments of your octagon are properly spaced. To do this, click the Extrusion tab in the Edit pane. Now you can click on each point and drag or numerically edit its location. Each point should have coordinates of +/- 12 or +/- 5 for *X* and *Z*.

Are you having trouble locating the points? Try clicking the **Wire Frame** button on the toolbar to make it easier to find and manipulate the point positions.

6. The default height of the shape we've drawn is 4 units. For our stop sign, we don't want the extrusion to be "thick," so we'll adjust the thickness by choosing the Extrusion tab in the Edit pane and changing the Height field from 4 to `0.1`. Now we have the thin octagon shape that will be the basis for our stop sign.



7. Next, we'll select the Color tab in the Edit pane and paint the shape red by selecting the "red" standard color.



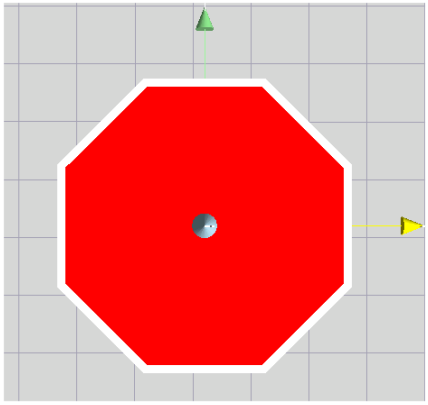
The next stage is to create the white border. For this, we need to copy and paste the red octagon, scale it a bit larger (say by 1.06), make the copy white, and make it even thinner than the red octagon (say 0.06 instead of 0.1). (If it is thinner, then only the outer white portion will show.)

1. First, let's do the copy and paste. That is as simple as selecting the octagon if it is not already selected, then choosing **Edit > Copy** then **Edit > Paste**. You can also use **CTRL+C** then **CTRL+V**. Once you have issued a Paste command, you get the familiar cross-hair pointer and can place the copy anywhere in the layout.

To move shapes on the drawing board, you must first click on the Transformation tab. With any other tab selected, the shape cannot be moved; zoom and rotation mouse movements can be performed with any tab selected.

2. Eventually we will want the two octagons to share the same location, but let's defer that to make them easier to manipulate separately.
   - Select the pasted octagon and, on the Extrusion tab, change the Height field to `0.06`.
   - On the Transformation tab, set the Scale to `1.06`.
   - Next, select the Color tab and make this octagon white.
   - Now click **Zero All** in the Transformation tab. Select the original (red) octagon and click **Zero All** for that one as well.

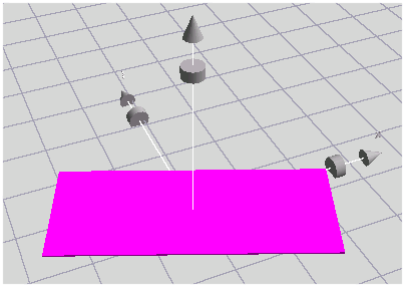
   Your shape should look very familiar now.



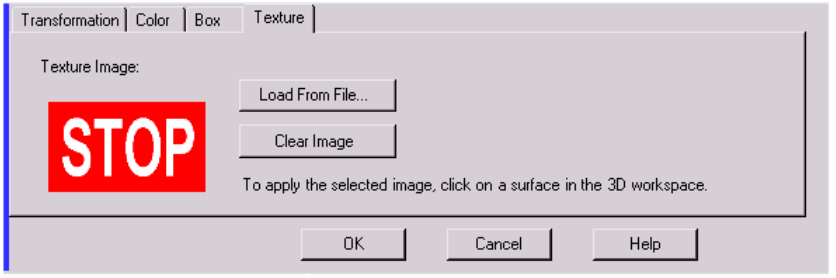
The final step is to get the word "**STOP**" onto the sign. There is no 3D text feature, so the solution is to use a bitmap picture. It's easy to create a white-on-red rectangular "STOP" bitmap in Paint or some other program, and save it as a *.bmp* file (*.bmp* files are not "lossy" like *.jpg* files, which are better used for pictures). We have provided this bitmap already as *stopsign.bmp*.

You cannot paste a bitmap onto the "end" of an extrusion, so we'll have to create one more shape – a box – and overlay it on top of the stop sign.

1. Press the **Box** button ( ) and click to place the shape on the drawing board.
   - On the Box tab, set the Width to **`24`**, Height to **`0.12`**, and the Depth to **`10`**.
   - On the Transformation tab, perform a **Zero All** and change the *Y* translation to **`0.06`**. (This *Y* translation, or else a bigger thickness, is necessary because a box by default has its top and bottom above and below the *XZ* plane, while an extrusion by default has its base on the *XZ* plane and does not extend below.)



2. Your box should be blue so it should be apparent when it is in the right place. Now click the Texture tab, choose **Load From File**, and choose the *stopsign.bmp* file. When you see the image in the tab, just click the top surface of the box, and you have your stop sign. You might see a thin blue "seam" around the edge of the box…if so, on the Color tab, make the box red to minimize this effect.



Presto! You have a stop sign!

Your sign is about 25.5 units across. Note the distance units before using this shape. If your distance unit is inches you are probably all set. Otherwise, you will have to scale the shape. (If you save the shape to your library, you should probably scale the shape definition by 1/12 because library shapes are all in feet for consistency. If you want the shape to be stored in meters, or your drawing is in meters, scale by 1/39.37.)

Tutorial 9: Revolutions

Revolutions come in handy for many types of non-cylindrical shapes such as tires and wheels, tanks or other containment vessels, and similar forms. This exercise shows how you can use revolutions creatively to make a coffee mug.

Your first impulse in designing a coffee mug in 3D might be to use a cylinder. After all, that is the shape of most mugs. However, a cylinder has many disadvantages. There is no way to show the "inside wall" of a standard cylinder in Arena 3DPlayer, so your cup would always be full to the brim (even if you used texture mapping to "color" the top surface only). The edges would be too sharp. And you would still need to draw a handle.

Revolutions answer all three of those concerns.

Your first step is to make the basic coffee mug shape. To do this, first visualize a coffee mug. Remove the handle and cut the mug in half vertically. Imagine the cut surface…it is basically a "U" shape. What you want to do is take half of that "U" and rotate it around the center axis of the cup.

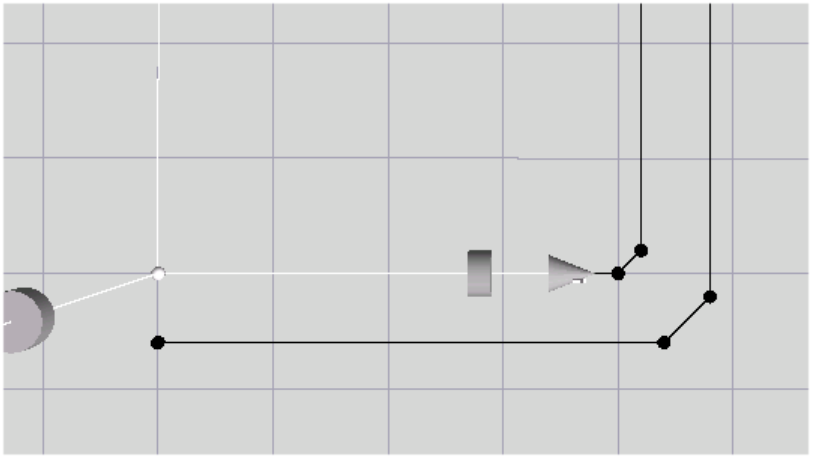
This time instead of starting with a static shape, let's start with a new library shape.

1. Go to the Shape Manager, choose **Misc** as the category, and click the **Add** button in the right-hand pane. You will see a blank, new library shape appear in the right-hand set of shapes.
2. Change the name of the new shape from **Root** to **`CoffeeMug`**. Then click the right-hand **Edit** button.

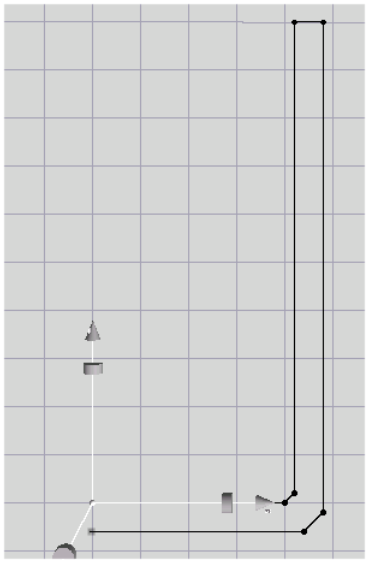
   This time when the Shape Editor opens, because we specified that we wanted to create a "new" shape, the editor is blank and there is no existing geometry.
3. Select **Snap to Grid** and then press the **Revolution** button ( ) from the Shape Editor toolbar.

   Instead of being concerned with exact dimensions, we'll base our "design" on proportion. (For a discussion of the relationship of units and distance, see "Distance units" on page 105.) The axis of rotation is the Z axis, so we will be drawing half of the cross-cut "U" shape.

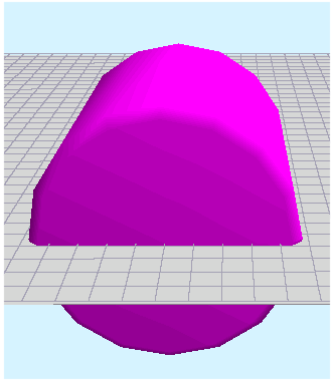
   - We'll draw our mug shape to be a little more than twice as high as it is wide, and we'll apply a little smoothing to the bottom inside and outside edges as we draw the "half-U" that we will rotate. (See the detail below.)
   - This time, since the pivot point is the Z axis, it is important that our first click be at the base point. Click at any intersection and move east 4 grid units (half the diameter of the mug bottom) and click. Smooth the corner and add points to define the mug shape. Use a mug height of about 10 grid lines and a mug "wall" of 3 snap increments.



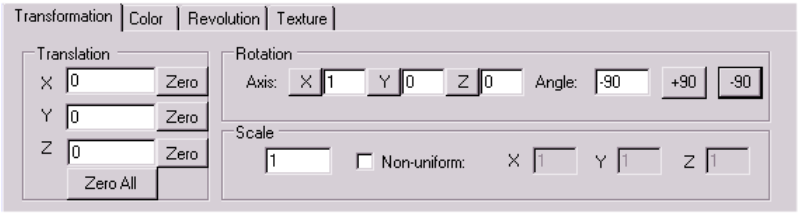
(View showing the drawing shape before clicking the point for the final segment of the polyline.)



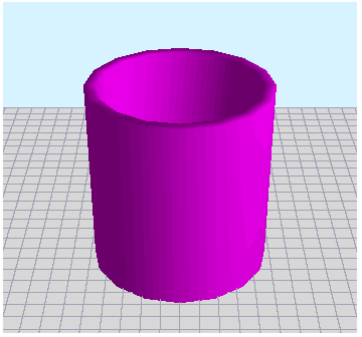
- Once we click again on the first point (the white dot at the base point) to close the loop, the revolution will appear as shown below.



4. At this point, the only problem is your coffee mug body is on its side. This is easy to fix. From the Transformation tab, click the ***X*** button (next to the word Axis) and click the ***-90*** button to stand the mug upright with the opening at the top.



Now your mug should look correct.



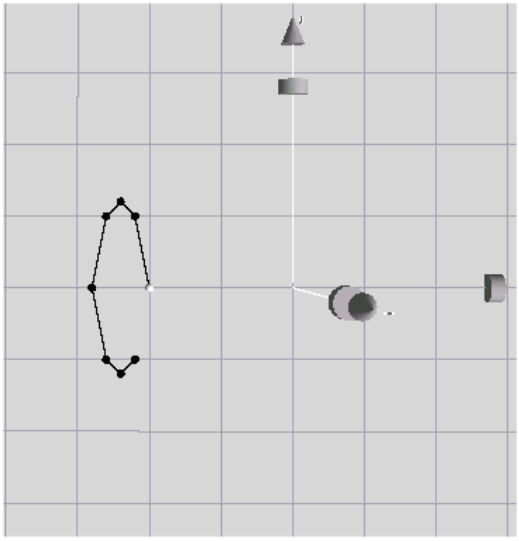
5. From the Transformation tab, click the **Zero All** button to center the mug body within the group that comprises the shape. That group in turn will be centered at the location where you eventually place the shape when using it from the library.

6. Next we need to attach a handle.

   This requires some thought. A mug handle is similar to a ring shape, but it is slightly elongated. Once we have drawn the handle, we can narrow the ring by non-uniform manipulation of the scale from the Transformation tab for the *X* axis (the loop of the handle). We'll get to that later; first let's draw the basic handle shape.

   - Remembering that we are working with a revolution from the Z axis, we need to define the shape of half of the cross-section of the handle. To define the size of the opening, we'll first click on an intersection to set the base point.
   - Now move two grid units west of the base point to begin drawing the handle cross section. (We'll modify the shape once the drawing is complete.) Your first click (white dot) will be at the intersection along the *X* axis horizontal line. From there

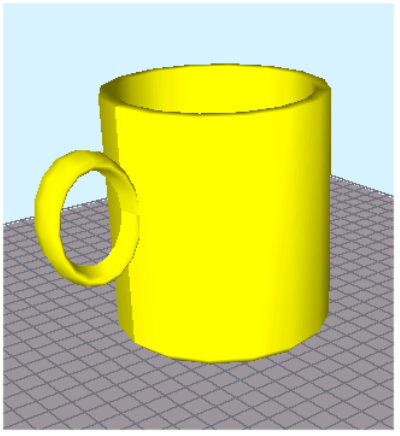
click points to resemble the drawing below. When you double-click to complete the polygon, the full shape of your drawing will form around the distance of your points from the base point. This distance is what creates the center of the loop.



7. The next step is to refine the shape to be an oval and to attach it to the mug. On the Transformation tab, check the "Non-uniform" box and change the *X* scale to **`0.75`**. Can you see how your handle is now elongated? If you feel that the handle is too wide, you can also change the *Z* scale to **`0.75`**.

   - Now move the handle to the edge of the mug, which you can do by dragging the *X*, *Y*, and *Z* cones on the ring's triad. (*Z* will be zero, so you can type in that value. If you have drawn your mug using our proportions, our handle is placed at *X*, 32; *Y*, 32; *Z*, 0.)

8. Finally, let's change the color by selecting a new color from the Color tab of the edit menu and applying it to the mug and handle...in this case, we chose yellow.



Keyboard Controls for Camera Movement

Keyboard Equivalents for Stepping the Viewpoint

Keystrokes	Movement
SHIFT+Right Arrow	Counter-clockwise rotation of the geometry in the *X*-*Z* plane
SHIFT+Left Arrow	Clockwise rotation of the geometry in the *X*-*Z* plane
SHIFT+Up Arrow	Upward tilt of the geometry
SHIFT+Down Arrow	Downward tilt of the geometry
Number Pad +	Zoom in
Number Pad -	Zoom out
Right Arrow	Right pan
Left Arrow	Left pan
Up Arrow	Forward movement
Down Arrow	Backward movement

Keyboard Equivalents for Continuous Movement of the Viewpoint

Keystrokes	Movement
CTRL+SHIFT+Right Arrow	Counter-clockwise rotation of the geometry in the *X*-*Z* plane
CTRL+SHIFT+Left Arrow	Clockwise rotation of the geometry in the *X*-*Z* plane
CTRL+SHIFT+Up Arrow	Upward tilt of the geometry
CTRL+SHIFT+Down Arrow	Downward tilt of the geometry
CTRL+Number Pad +	Zoom in
CTRL+Number Pad -	Zoom out
CTRL+Right Arrow	Right pan
CTRL+Left Arrow	Left pan
CTRL+Up Arrow	Forward movement
CTRL+Down Arrow	Backward movement
ESC	Stop movement

Index

Numerics

3D enhancements ■ 97
3D objects
  adding ■ 20
  globals ■ 23
  queues ■ 22
  resources ■ 20
  storages ■ 24
3D pane ■ 16, 19
  editing ■ 33
  floor ■ 19, 37
  grid ■ 19, 37
  navigation ■ 11
  origin triad ■ 19
  toolbar ■ 7
  understanding the navigation ■ 11
3D wall systems ■ 56

A

Accumulating Distance ■ 21
animation
  analyzing ■ 96
  building a simple example ■ 31
  capturing with AVI ■ 99
  changing the run speed ■ 13
  creating 2D in Dashboard ■ 61
  importing information from Arena ■ 91
  live components ■ 97
  opening an existing ■ 10
  redefining the time frame ■ 13
  running ■ 13
  static components ■ 98
animation information ■ 91
animation methods ■ 15
animation object
  selection ■ 34
Arena 3DPlayer environment ■ 5
Arena settings
  Generate Playback File ■ 31
AVI Capture ■ 99
AVI files ■ 99, 101
AVI viewpoint ■ 100

B

binary format ■ 31

C

CAD layouts
  importing ■ 54
clocks
  analog ■ 62
  digital ■ 61
Color tab ■ 19, 36
contact information ■ 3
Customer Support Center ■ 2

D

Dashboard ■ 16
  adding objects ■ 61
  editing objects ■ 70
  import information ■ 93
  Named Views ■ 12
  navigation ■ 59
  toolbar ■ 8
  using the 2D pane ■ 59
Dashboard objects
  clocks ■ 61
  histograms ■ 69
  images ■ 64
  levels ■ 65
  plot ■ 67
  text ■ 66
  variables ■ 62
distances ■ 78
document conventions ■ 1
duplicate shapes ■ 45
DXF files ■ 54
  importing ■ 55

E

Edit pane ■ 19
editing in 3D pane ■ 33
editing objects
  Dashboard ■ 70

entity ▪ 43
entity pictures ▪ 43
extrusions ▪ 107

F

floor ▪ 32, 37
fonts ▪ 66
free-path transporters ▪ 76
  distances ▪ 78

G

Generate Playback File
  Arena settings ▪ 31
global shape animation ▪ 24
globals ▪ 23
grid ▪ 19, 37
grid settings
  3D pane ▪ 38
  dashboard ▪ 59
Grid Settings dialog ▪ 38
  Grid Spacing ▪ 38
  Snap Spacing ▪ 38
group selecting and editing ▪ 39
guided transporters ▪ 76
  networks ▪ 80

H

histogram ▪ 69

I

individual unit shape sets ▪ 77
inserting points on a path ▪ 89
intersections ▪ 77, 83

K

keyboard controls for camera movement ▪ 119

L

Layout Editor ▪ 16
layout file ▪ 9
line-type residence paths ▪ 86
  Accumulating Distance ▪ 86
live components ▪ 97

M

manipulating shapes
  3D pane ▪ 33
material handling ▪ 73
  path directionality ▪ 81
  segments ▪ 75
menus and commands ▪ 6
mode
  Demo ▪ 16
  Run ▪ 13
movie files ▪ 99
  recording quality ▪ 101
moving the viewpoint ▪ 119
multiple selection
  rotation ▪ 39
  translation ▪ 39

N

Named Views ▪ 12
navigating in the 3D pane ▪ 11, 13
  "move the geometry" paradigm ▪ 11
network links ▪ 80, 81
networks ▪ 80

O

object
  adding 2D ▪ 61
  rotation ▪ 35
  scaling ▪ 36
online help ▪ 2
origin triad ▪ 19, 34

P

parking areas ▪ 84
paths ▪ 73, 81
  deleting ▪ 88
  drawing ▪ 85
  editing ▪ 88
  editing points ▪ 88
  guide type (routes, segments, distances, networks) ▪ 85
  inserting points ▪ 89
  markers (anchor point) ▪ 85
  markers (endpoint/midpoint) ▪ 85

moving ■ 88
residence-type (queues, seize areas, storages, parking areas) ■ 85
phone support ■ 2
playback file ■ 9
sharing with others ■ 16
plot ■ 67
point-type residence paths ■ 86, 87
primitive ■ 49

Q

queues ■ 22
modifying ■ 95

R

recording quality ■ 101
resources ■ 20
refining ■ 94
seize areas ■ 21
revolutions ■ 113
ride points ■ 78
attaching and editing ■ 78
rotating an object ■ 35
rotation ■ 35
rotation of a multi-selection ■ 39
routes
importing from Arena ■ 74
Run mode ■ 13
run speed ■ 13
running the animation ■ 13

S

scaling an object ■ 36
segment set ■ 74
Seize Area dialog
Accumulating Distance ■ 21
seize areas ■ 21
modifying ■ 95
shape assignment
selecting new shapes ■ 44
shape libraries ■ 42, 45
Shape Manager ■ 41
Graphics List ■ 41
Tree View ■ 41
shapes
adding multiple identical copies ■ 45
drawing your own ■ 47
imported ■ 53
importing ■ 46
modifying ■ 95
saving your files ■ 98
standard (standard.slb) ■ 46
user-defined (user.slb) ■ 46
shared shape sets ■ 77
simulation information ■ 91
static components ■ 98
static shapes
from Arena ■ 56
static systems ■ 53
3D wall units ■ 56
adding shapes ■ 56
how to create ■ 54
stations ■ 77, 80
status expressions ■ 63
storages ■ 24

T

target shape ■ 43
text format ■ 31
Text object
fonts ■ 66
texture ■ 50, 112
time jumping during the animation run ■ 13
Toggle Split Screen ■ 20
Transformation tab ■ 19, 35
translation ■ 35
translation of a multi-selection ■ 39
transporters
adding ■ 76
free-path ■ 76
guided ■ 76
intersections ■ 76, 83
networks ■ 76
parking areas ■ 84
ride points ■ 78
stations ■ 77, 82
Tree View pane ■ 16, 17
visual cues ■ 18
trigger value ■ 24

Tutorial 1. Opening an existing animation ■ 10
Tutorial 2. Navigating in 3D ■ 13
Tutorial 3. Running the animation ■ 14
Tutorial 4. Building a simple animation ■ 31
Tutorial 5. Enhancing the simple animation ■ 47
Tutorial 6. Editing the Dashboard display ■ 71
Tutorial 7. Importing and running an Arena animation ■ 94
Tutorial 8. Drawing Extrusions ■ 107
Tutorial 9. Drawing Revolutions ■ 113

V

View control options ■ 11
viewpoint ■ 100
  varying the position ■ 100
VRML files ■ 51
  importing ■ 52

W

Web support ■ 2